StringStudio

USER MANUAL

A|A|S

Applied Acoustics Systems

Information in this manual is subject to change without notice and does not represent a commitment on the part of Applied Acoustics Systems DVM Inc. The software described in this manual is furnished under a license agreement. The software may be used only in accordance of the terms of this license agreement. It is against the law to copy this software on any medium except as specifically allowed in the license agreement. No part of this manual may be copied, photocopied, reproduced, translated, distributed or converted to any electronic or machine-readable form in whole or in part without prior written approval of Applied Acoustics Systems DVM Inc.

Copyright ©2014 Applied Acoustics Systems DVM Inc. All rights reserved. Printed in Canada.

Program Copyright ©2005-2014 Applied Acoustics Systems, Inc. All right reserved.

String Studio VS is a Trademark of Applied Acoustics Systems DVM Inc. Windows and Windows Vista are registered trademarks of Microsoft Corporation in the United States and other countries. Mac OS and Audio Units are registered trademarks of Apple Corporation. VST Instruments and ASIO are trademarks of Steinberg Soft Und Hardware GmbH. RTAS is a registered trademark of Avid Technology Inc. All other product and company names are either trademarks or registered trademarks of their respective owner. Unauthorized copying, renting or lending of the software is strictly prohibited.

Visit Applied Acoustics Systems DVM Inc. on the World Wide Web at

www.applied-acoustics.com

Contents

1 Introduction **7**

1.1 System Requirements 7

1.2 Installation 8

1.3 Authorization and Registration 8

1.3.1 Your Computer is Online 8

1.3.2 Your Computer is Offline 9

1.4 Getting Started 10

1.4.1 Using *String Studio VS* in Standalone Mode 10

1.4.2 Exploring the Factory Sounds 11

1.4.3 Using *String Studio VS* as a Plug-in 12

1.5 Getting Help 12

1.6 About this Manual 12

2 Architecture of *String Studio VS* **13**

2.1 General Signal Flow 13

2.2 Interface 14

2.2.1 The Play View 14

2.2.2 The Edit View 15

2.2.3 The FX view 16

3 Bank and Program Management **17**

3.1 Banks and Programs 17

3.2 Saving Programs 17

3.3 The Bank Manager 17

3.4 Using MIDI Bank and Program Changes 19

3.5 Backups of Banks and Programs 19

3.6 Exchanging Banks and Programs 19

3.7 Restoring the Factory Library 20

3.8 Importing Programs from String Studio VS-1 20

4 Parameters **22**

4.1 General Functioning of the Interface 22

4.1.1 Knobs 22

4.1.2 Switches 22

4.1.3 Drop-down Menus 22

4.1.4 Modulation Signals 22

4.1.5 Synchronisation 23

4.2 The Edit View 23

4.2.1 The String Module 23

4.2.2 The Exciter Module 24

4.2.3 The Body Module 27

4.2.4 The Damper Module 28

4.2.5 The Termination Module 28

4.2.6 The Geometry Module 29

4.2.7 The Distortion module 30

4.2.8 The Filter Module 31

4.2.9 The Envelope Module 34

4.2.10 The LFO Module 35

4.3 The FX View 36

4.3.1 EQ 37

4.3.2 Compressor 38

4.3.3 Delay 40

4.3.4 Distortion 40

4.3.5 Chorus 41

4.3.6 Flanger 42

4.3.7 Phaser 45

4.3.8 Wah 47

4.3.9 Notch Filter 47

4.3.10 Reverb 48

4.4 The Play View 50

4.4.1 The Clock Module 50

4.4.2 The Key Module 50

4.4.3 Unison . . . 51
4.4.4 The Glide Module . . . 51
4.4.5 The Vibrato Module . . . 51
4.4.6 The Arpeggiator Module . . . 52
4.4.7 Pitch Wheel . . . 53
4.4.8 Modulation Wheel . . . 54
4.4.9 Ribbon . . . 54

5 Utility Section 55

5.1 The MIDI LED . . . 55
5.2 Tuning . . . 55
5.3 Polyphony . . . 56
5.4 History and Compare . . . 56
5.5 Volume . . . 56
5.6 Level Meter . . . 56
5.7 The About Box . . . 57

6 Audio and MIDI Settings 58

6.1 Audio Configuration . . . 58
6.1.1 Selecting an Audio Device . . . 58
6.1.2 Latency . . . 58
6.2 MIDI Configuration . . . 59
6.2.1 Selecting a MIDI Device . . . 59
6.2.2 Creating MIDI Links . . . 59
6.2.3 Creating a default MIDI Map . . . 59
6.2.4 MIDI Program Changes . . . 60
6.2.5 MIDI Bank Changes . . . 60
6.2.6 Pitch bend . . . 60
6.2.7 Modulation wheel . . . 60

7 Using *String Studio VS* as a Plug-In **61**

7.1 Audio and MIDI Configuration 61

7.2 Automation 61

7.3 Multiple Instances 61

7.4 Saving Projects 61

7.5 Performance 61

8 License Agreement **62**

1 Introduction

String Studio VS is a synthesizer dedicated to the emulation of string instruments. The synthesizer is entirely based on the AAS physical modeling technology and uses no sampling nor wave tables. Instead it produces sound by solving, on the fly, mathematical equations modeling the different components involved in string instruments and how they interact. This elaborate synthesis engine responds dynamically to the control signals it receives while you play thereby reproducing the richness and responsiveness of real string instruments.

String Studio VS features three types of exciters (hammer, pick, and bow) an accurate model of a string, a model of the fret/finger interaction, a damper model and different types of soundboards. The combination of these different elements allows for the reproduction of a wide range of string instruments. *String Studio VS* is also equipped with a distortion module, filters and a comprehensive output effect stage to add the finishing touch to the sound. Finally, String Studio offers a wide range of performance features, including keyboard modes, portamento, vibrato and legato functions, a programmable pattern arpeggiator, and a complete set of MIDI features for optimal controller integration.

Before discussing the synthesizer in more detail, we would like to take this opportunity to thank you for choosing an AAS product. We sincerely hope that this product will bring you inspiration, pleasure and fulfill your creative needs.

1.1 System Requirements

The following minimum computer configuration is necessary to run *String Studio VS*:

Mac OS :

- Mac OS X 10.6 or later
- Intel Core processor or later
- 512 MB of RAM
- 70 MB of free hard drive space
- 1024 x 768 screen resolution
- Built-in audio interface

Windows :

- Windows XP SP2 32-bit or Windows Vista/7 32-bit/64-bit
- Intel Core or equivalent processor
- 512 MB of RAM
- 70 MB of free hard drive space
- 1024 x 768 screen resolution
- Windows-compatible audio interface

- Windows-compatible MIDI interface/keyboard

Keep in mind that the computational power required by *String Studio VS* depends on the number of voices of polyphony and the sampling rate used. These computer configurations will enable you to play the factory sounds with a reasonable number of voices but performances will vary depending on your specific computer configuration.

1.2 Installation

Simply double-click on the installer file that you have downloaded and follow the instructions of the installer.

1.3 Authorization and Registration

String Studio VS uses a proprietary challenge/response copy protection system which requires authorization of the product. A *challenge code* is a long string of capital letters and numbers that is generated uniquely for each machine during the registration process. The *response code* is another unique string of capital letters and numbers generated from the data encrypted in the challenge code. As the keys are unique to each machine, it is necessary to go through this procedure every time the program is installed on a new computer.

Note that it is possible to use the program during 15 days before completing the authorization process. After that period, the program will not function unless it is authorized.

1.3.1 Your Computer is Online

The authorization process is very simple if your music computer is connected to the internet since *String Studio VS* program will connect to the AAS server and take care of the key exchange automatically.

After starting the application, a message will appear telling you that the application needs to be authorized as shown in Figure 1. Enter your serial number and click on the *Authorize* button. The program will then connect to the AAS server and complete the authorization process.

If this is the first AAS product that you authorize on your computer, or if no registration information can be related to your serial number by our server, you will be asked to provide your name and email address for registration purposes. Note that only a valid email address is required to register your product. Registration of your product will entitle you to receive support and download updates when available, as well as take advantage of special upgrade prices offered from time to time to registered AAS users.

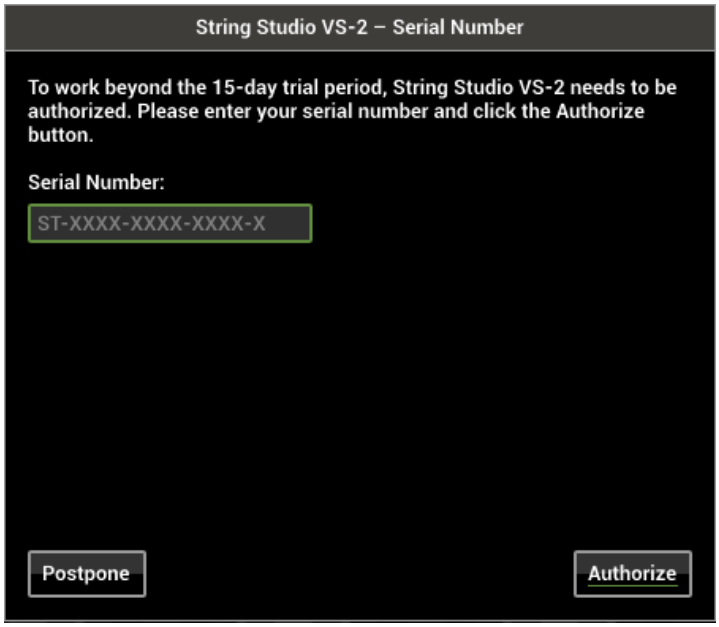

String Studio VS-2 – Serial Number
To work beyond the 15-day trial period, String Studio VS-2 needs to be authorized. Please enter your serial number and click the Authorize button.
Serial Number:
ST-XXXX-XXXX-XXXX-X
Postpone
Authorize

Figure 1: Online Authorization.

1.3.2 Your Computer is Offline

If your music computer is not connected to the internet you will need to obtain the *response code* from an internet connected computer or by contacting AAS.

After starting the application, a message will appear telling you that the application needs to be authorized. After clicking on the *Authorize* button, a pop-up window will appear as shown in Figure 1. Enter your serial number and click on the *Authorize* button. The program will then inform you that your computer is not connected to the internet, click on the *Offline Options* button and a new pop-up window will appear as shown in Figure 2.

Your serial number as well as the automatically generated *challenge code* are displayed but you need to obtain the *response code*. To do so, take note of your *serial number* and *challenge code* and proceed to an internet connected computer. Launch your browser and go to the unlock page of the AAS website located at:

http://www.applied-acoustics.com/unlock/

Enter your serial number and *challenge code* in the form, follow the instructions, and the *response code* will appear on screen. Write it down, go back to your music computer, and enter the *response code* in the authorization pop-up window. This will complete the authorization procedure.

If you prefer, you can also contact us by email at support@applied-acoustics.com with your *serial number* and *challenge code* and we will send you back your *response code*.

Should you not have access to the internet, AAS support representatives are available to assist

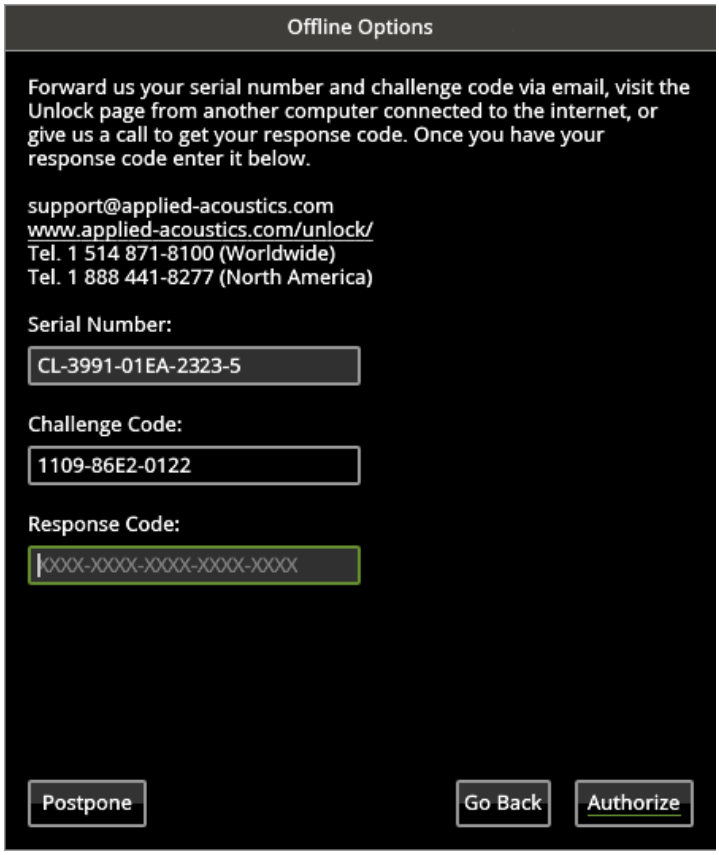

Offline Options
Forward us your serial number and challenge code via email, visit the Unlock page from another computer connected to the internet, or give us a call to get your response code. Once you have your response code enter it below.
support@applied-acoustics.com
www.applied-acoustics.com/unlock/
Tel. 1 514 871-8100 (Worldwide)
Tel. 1 888 441-8277 (North America)
Serial Number:
CL-3991-01EA-2323-5
Challenge Code:
1109-86E2-0122
Response Code:
XXXX-XXXX-XXXX-XXXX-XXXX
Postpone
Go Back
Authorize

Figure 2: Offline Authorization.

you in the unlock and registration process Monday to Friday, 9am to 6pm EST. You may contact us by phone at:

- North America Toll-free number: 1-888-441-8277
- Outside North America: 1-514-871-8100

1.4 Getting Started

1.4.1 Using *String Studio VS* in Standalone Mode

String Studio VS comes with a standalone versions allowing you to play it without having to open your sequencer. This can be convenient to explore *String Studio VS* and its library, play it live or do some sound design work. To start *String Studio VS* in standalone mode, simply follow the instructions below:

- **Windows** - Double-click on the *String Studio VS* icon located on your desktop or select *String Studio VS* from the **Start** > **All Programs** > menu.
- **Mac OS** - Double-click on the *String Studio VS* icon located in the Applications folder.

Before you start exploring the program, take a moment to set up you audio and MIDI configuration as explained below.

Audio and MIDI Configuration

Audio and MIDI configuration tools are available by clicking on the **Audio Setup** button located in the lower left corner of the *String Studio VS* interface. The **Audio Setup** dialog first allows you to select an audio output device from those available on your computer. Multi-channel interfaces will have their outputs listed as stereo pairs.

On Windows, the audio output list is organized by driver type. The device type is first selected from the *Audio Device Type* drop-down list. If you have ASIO drivers available, these should be selected for optimum performance. The **Configure Audio Device button** allows you to open the manufacturer's setup program for your audio interface when available.

Once the audio input has been selected, you can then select a sampling rate and a buffer size from those offered by your audio interface.

The list of available MIDI inputs appears at the bottom of the dialog. Click on the checkbox corresponding to any of the inputs you wish to use.

1.4.2 Exploring the Factory Sounds

String Studio VS comes with a wide range of factory programs right out of the box which amounts to a huge range of sounds before you have even turned a single knob. As you would expect, the best way of coming to grips with the possibilities *String Studio VS* offers is simply to go through the programs one at a time.

String Studio VS uses the notions of *Banks* and *Programs* to organize and classify sounds. A program or preset is a stored set of parameters corresponding to a given sound. The programs are grouped and organized in banks.

The name of the currently loaded bank and program are displayed at the top of the interface. One navigates among the different banks and programs by using the arrows in each of the corresponding boxes or by opening the associated drop-down menu by clicking inside these boxes. Banks and programs are managed using the *Bank Manager* which is revealed by clicking on the *Manage* button appearing above the right-top corner of the *Bank* box. Playing programs and organizing them is pretty straightforward, please refer to Chapter 3 for a complete description of the bank and program management operations.

1.4.3 Using *String Studio VS* as a Plug-in

String Studio VS integrates seamlessly into the industry's most popular multi-track recording and sequencing environments as a virtual instrument plug-in. *String Studio VS* works as any other plug-in in these environments so we recommend that you refer to your sequencer documentation in case you have problems running *String Studio VS* as a plug-in. Note that in plug-in mode the audio and MIDI inputs, sampling rate, and buffer size are determined by the host sequencer.

1.5 Getting Help

AAS technical support representatives are on hand from Monday to Friday, 9am to 6pm EST. Whether you have a question on *String Studio VS*, or need a hand getting it up and running as a plug-in in your favorite sequencer, we are here to help. Contact us by phone or email at:

- North America Toll Free: 1-888-441-8277
- Worldwide: 1-514-871-8100
- Email: support@applied-acoustics.com

Our online support pages contain downloads of the most recent product updates, and answers to frequently asked questions on all AAS products. The support pages are located at:

www.applied-acoustics.com/support/

1.6 About this Manual

Throughout this manual, the following conventions are used:

- Bold characters are used to name modules, commands and menu names.
- Italic characters are used to name controls on the interface.
- Windows and Mac OS keyboard shortcuts are written as Windows shortcut/Mac OS shortcut.

2 Architecture of *String Studio VS*

2.1 General Signal Flow

String Studio VS is a synthesizer built around a **String** module which acts a a sound source. The general architecture of the synthesizer is presented in Figure 3 and follow the functioning of a real string instrument.

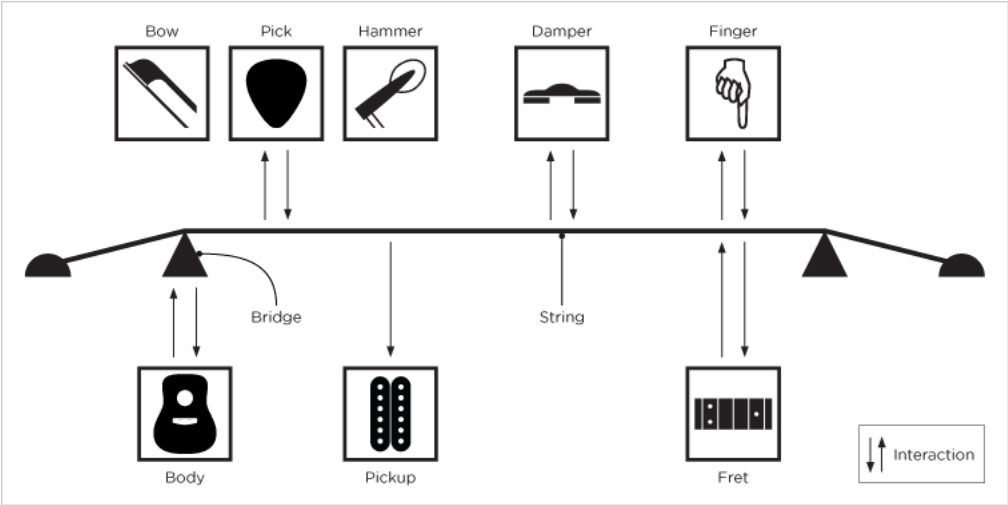


Figure 3: General signal flow of *String Studio VS*.

It is the vibration from the **String** which constitutes the main sound production mechanism of the instrument. The string is set into motion by the action of an **Exciter** which can be a hammer, a pick or a bow. The frequency of the oscillation is determined by the effective length of the string which is controlled by the finger/fret interaction or **Termination**. A **Damper** can be applied on the strings in order to reduce the decay time of the oscillation. This is the case on a piano, for example, when felt is applied on the strings when the keys and the sustain pedal are released. The vibration from the string is then transmitted to the **Body** of the instrument which can radiate sound efficiently. In some instruments, the string vibration is transmitted directly through the bridge. In other instruments, such as the electric guitar, a **Pickup** is used to transmit the string vibration to an amplifier.

In addition to these main modules, a **Filter** module has been included between the **String** and **Body** module in order to expand the sonic possibilities of the instrument. A **Distortion** module and an output effect stage complete the synthesizer as illustrated in the signal flow diagram of Figure 4.

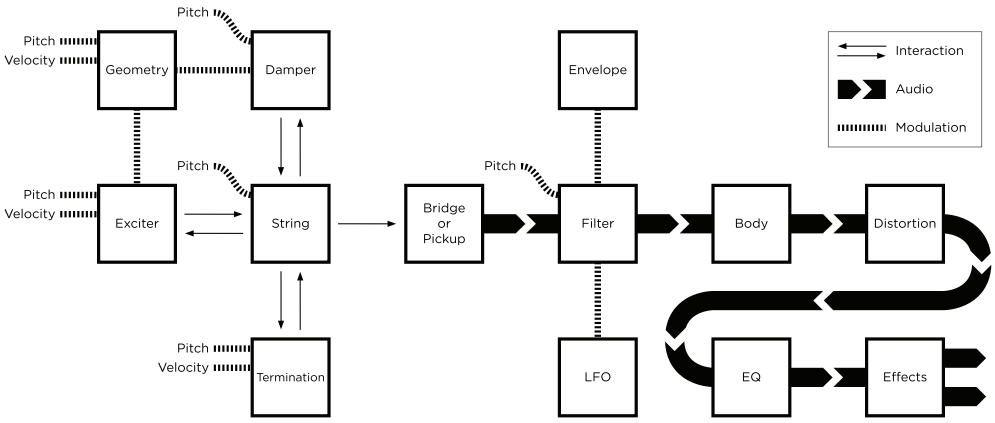


Figure 4: Signal flow of *String Studio*. Modulation signals: dotted lines.

2.2 Interface

The graphical user interface has been organized around three different views as shown in Figures 5, 6 and 7.

The first view, called the *Play* view of the instrument, gives access to different performance parameters as well as to a step sequencer. The second and third views, called the *Edit* and *FX* views respectively, are used for in-depth editing of the synthesis and effect parameters.

One can switch from one view to the other by using the *Play*, *Edit* and *FX* buttons located in the *Utility* section at the top of the interface. This section of the interface is common to all the views and includes the bank manager, used to access and manage sounds, as well as general settings and indicators. These tools are described in details in Chapter 3 and Chapter 5 respectively.

2.2.1 The Play View

The lower section of this view includes a master clock, keyboard, unisson, glide, vibrato and arpeggiator modules which will be described in more details in Chapter 4.

On the left of these parameters, one finds a pitch bend wheel and a modulation wheel. The modulation wheel is normally used to control the amount of vibrato in the sound but it can also be used to adjust any other parameter through MIDI links which will be described in Chapter 6. Just below is a clickable eight octave ribbon allowing one to play different notes on the range of the piano which can be useful when no MIDI keyboard is connected to the computer.

The middle section of this view allows one to turn the effects from the multi-effects module, compression and equalizer *on* and *off* and to rapidly adjust their main parameters.

Figure 5: The Play view.

2.2.2 The Edit View

The *Edit* view gives access to the synthesis parameters described in details in Chapter 4 and allows one to really go under the hood. One can switch between different synthesis modules using the buttons appearing at the bottom of this view.

Figure 6: The Edit view.

2.2.3 The FX view

The *FX* view includes an equalizer, a compressor a multi-effects, and a reverb module. The multi-effects module consists in two effects in series. The effect list includes a delay, distortion, chorus, flanger, phaser, wah wah, auto wah and a notch filter. The functioning of the effect modules is described in details in Chapter 4.

Figure 7: The FX view.

3 Bank and Program Management

String Studio VS comes with several factory presets, called *programs*, covering a wide range of sounds. This collection of programs lets you play and familiarize yourself with this synthesizer without having to tweak a single knob. Soon, however, you will be experimenting and creating your own sounds and projects that you will need to archive or exchange with other users. In this section, we review the management of programs.

3.1 Banks and Programs

Sounds are stored in banks contaning so-called *programs*. The name of the currently selected bank is shown in the *Bank* drop-down display located at the top of the *String Studio VS* interface. The list of available banks is viewed by clicking on the *Bank* display. A bank can be selected by navigating in the list of banks using the left and right-pointing arrows in the display or by clicking on its name when the list of banks is open.

The list of programs included in the currently selected bank can be viewed by clicking on the *Program* display located below the *Bank* display. A program is selected by using the left and right-pointing arrows or by clicking directly on its name. Once a program is selected, the value of the different parameters of the synthesizer are updated and it can then be played.

3.2 Saving Programs

Programs are saved by clicking on the *Save* button located on the top of the *Program* display. When a program has just been loaded, this command is greyed and therefore inactive. It is activated as soon as a parameter of the interface is modified. Clicking on this command replaces the stored version of the program with the new one.

The **Save As** command is activated by clicking on the corresponding button which opens the **Save Program** pop-up window. It is then possible to save the program under a new name or its current one in any of the available program banks. Note that if the original name of the program is used, a new program with the same name will be created at the end of the program list meaning that the original program is not erased. This also implies that it is possible to have many programs with the same name in the same bank.

3.3 The Bank Manager

Banks and Programs can be edited using the **Bank Manager**. The manager window is displayed by clicking on the *Manager* button located above the *Bank* display. It is closed by clicking again on the same button. On the left of the window, one finds the list of banks. Clicking on a bank name fills the list of programs located in the center of the window with the name of these included in the selected bank.

Figure 8: Bank and program manager window.

A new bank can be created by clicking on the + button below the bank list. This opens the **Create New Bank** window in which the name of the new bank can be entered. A bank can be deleted by first selecting it in the bank list and then clicking on the - button. Be careful, this command erases a bank and all the programs it contains; this operation is permanent and can not be undone. In order to rename a bank, simply click on the *Rename* button and enter a new name.

Banks and the information corresponding to each of its programs is stored in a simple text file on your computer hard disk. In order to view these bank files, click on the *Show Files* button under the bank list. On Windows, this command will open an Explorer window at the location where the files are stored. On Mac OSX, the command has a similar effect and opens a Finder window. All the bank file names follow the same format and begin with the bank name. These files can be used for backups or to exchange presets with other users.

The list of programs included in the selected bank is displayed in the program list in the center of the manager window. Presets are selected by clicking on their name which updates the program information appearing on the right of the preset list. Program information includes the name of the preset, its author and comments. This information can be updated by clicking on the corresponding box which opens an edition window. Note that multiple presets can be updated simultaneously by selecting more than one preset at once and clicking on a preset information box.

A multiple selection consisting of adjacent programs is obtained by holding down the *Shift* key on the computer keyboard and then clicking on the name of the first program to be copied and then the last one. A non-adjacent multiple selection is obtained by holding down the *Ctrl/command* computer key and clicking on the name of the different programs to be copied. It is also possible to select all programs at once by clicking on the *Select All* button at the bottom of the program list.

Programs can be copied to another bank by clicking on the *Copy* button. A program must first be selected by clicking on its name on the program list; it is then copied by moving the mouse to a given bank in the *Bank* list on the right and clicking on the bank name. The **Move** command is activated by clicking on the *Move* button; it copies a preset to a new bank but also erases it in the

original bank. A multiple selection of programs can be used with the **Copy** and **Move** commands

Programs can be deleted from a bank by first selecting them and then clicking on the *Delete* button. This will move the programs to a special bank called *Trash* which is located below the regular list of banks. This means that deleted programs can always be recuperated as long as they are not deleted from the *Trash* bank. The content of the *Trash* bank is viewed by clicking on its name; the different programs can then be moved to the other banks as explained above. The *Trash* bank can be emptied by clicking on the *Empty Trash* button which appears below the program list when the *Trash* bank is selected. Be careful as this command can not be undone.

3.4 Using MIDI Bank and Program Changes

Banks and programs can be changed using MIDI bank and program change commands. For more information on how to use these commands, please refer to sections 6.2.4 and 6.2.5.

3.5 Backups of Banks and Programs

User banks are stored on disk as simple text files located in the following folders:

On **Mac OS**:
/Users/[user name]/Library/Application Support/Applied Acoustics Systems/String Studio VS-2/Banks

On **Windows**:
%AppData%\Applied Acoustics Systems\String Studio VS-2\Banks

The bank files saved by *String Studio VS* are named using the following convention:

[name of bank].VS-2 Bank

These file contain all the information corresponding to the programs they include. These files can be displayed directly from *String Studio VS* by opening the *Bank* manager and clicking on the *Show Files* button. This will open an Explorer or Finder window on Windows or Mac OS respectively at the right location.

The simplest way to create a backup of banks and programs is to make a copy on an external media of the above mentioned folders. Individual banks can be backed-up by making copies of individual bank files.

3.6 Exchanging Banks and Programs

Banks and programs can easily be shared with other *String Studio VS* users. This operation simply involves the exchange of the above mentioned user bank files. When a new bank file is copied to the bank folder, it is automatically available to *String Studio VS*.

Note that individual programs can not be exported. They always appear inside a bank file. If you only wish to share a few programs, create a new bank, copy the programs you wish to exchange to this bank and share the corresponding bank file.

3.7 Restoring the Factory Library

If necessary, it is possible to restore the original factory library of banks and programs. The original factory bank files are located in the following folders:

On **Mac OS** startup disk:
/Library/Application Support/Applied Acoustics Systems/String Studio VS-2/Factory Library

On **Windows 64-bit**:
C:\Program Files (x86)\Applied Acoustics Systems\String Studio VS-2\Factory Library

On **Windows 32-bit**:
C:\Program Files\Applied Acoustics Systems\String Studio VS-2\Factory Library

Restoring the factory library simply involves copying the files contained in these folders and pasting them in the user bank folders listed in Section 3.5. The user bank folders can be opened directly in an Explorer or Finder window, on Windows and Mac OS respectively, or by using the *Show Files* command directly from the *String Studio VS* bank manager.

Note that if you have bank files with the original factory bank names in your user bank folder, they will be replaced by the original factory files. This means that you will lose programs that you would have modified or created in these banks. This operation must therefore be done with caution and it is recommended that you make copies or rename your user banks before proceeding with the restore.

3.8 Importing Programs from String Studio VS-1

String Studio VS-2 includes a converter that allows one to import programs from *String Studio VS-1*. The conversion operation first involves the creation of an *String Studio VS-1* .axf export file using the **Export** command from the **File** menu of *String Studio VS-1* . This file must then be copied with the other *String Studio VS-2* banks in the following folder:

On **Mac OS**:
/Users/[user name]/Library/Application Support/Applied Acoustics Systems/String Studio VS-2/Banks

On **Windows**:
%AppData%\Applied Acoustics Systems\String Studio VS-2\Banks

The simplest way to access the bank folder is to use the *Show Files* button in the *String Studio VS-2* bank manager which will open a Finder or Explorer window on Mac OS X or Windows respectively at the right location.

The *String Studio VS-1* export file will then automatically be converted to an *String Studio VS-2* bank having the same name as that of the .axf export file. This new bank will appear along with the other banks in the banks manager and include converted versions of all the programs contained in the .axf export file. Note that the programs will appear in the new bank as a flat list which means that any sub-folder organization within the export file will be lost in the conversion process.

While the great majority of programs should be recuperated without noticeable differences, the conversion program is not infallible which means that some programs might need some readjustments after the conversion. This is due to the fact that the mapping of the parameters from *String Studio VS-1* to *String Studio VS-2* is not direct as a result of changes in the architecture, modules and the effect section between the two versions.

4 Parameters

This section can be used as a reference for the different controls appearing on the *String Studio VS* graphical interface. We begin by describing the behavior of the different types of controls appearing on the interface and then describe the parameters of each module of the synthesizer.

4.1 General Functioning of the Interface

4.1.1 Knobs

The synthesizer parameters are adjusted using controls such as knobs, switches and numerical displays. A specific control is selected by clicking on it. A coarse adjustment is obtained by click-holding the parameter and moving the mouse, or the finger on a track pad, either upwards and downwards or leftwards and rightwards. The value of the parameter replaces its label while it is being adjusted.

Fine adjustment of a control is obtained by holding down a modifier key of the computer keyboard (Shift, Ctrl, Command or Alt key) while adjusting the parameter.

Double clicking on a knob brings it back to its default value when available.

4.1.2 Switches

Switches are turned *on* or *off* by clicking on them. They are used to activate or deactivate modules and the *sync* feature of some parameters.

4.1.3 Drop-down Menus

Some displays reveal a drop-down menu when clicking on them. Adjustment of the control is obtained by clicking on a selection.

4.1.4 Modulation Signals

Different parameters can be modulated by the signal from different sources including the MIDI keyboard, envelope generators and the **LFO** modules. Modulation signals are controlled with small gain knobs located below the corresponding modulated parameters.

The *Key* modulation knobs are used to modulate a parameter depending on the note played on the keyboard. When in its center position, the value of the corresponding parameter is equal across the whole range of the keyboard. Turning the knob to the left increases the value of the parameter for low notes while decreasing its value for high notes. The variations are applied relative to the middle C (C3, MIDI note 60) whose value is always that corresponding to the settings of the actual

parameter knob. Turning the modulation knob to the right has the opposite effect and increases the value of the parameter for high notes while decreasing it for low notes.

The *Vel* modulation knobs are used to modulate the value of a parameter depending on the velocity signal received from the keyboard so that the value of a parameter increases as notes are played harder on the keyboard. The position of the knob is used to adjust the amount of modulation applied to the parameter. In its leftmost position, the modulation source is turned *off* and the value of the parameter does not vary with the velocity signal from the keyboard. Turning the knob clockwise increases the effect of the modulation signal on the value of the parameter.

Modulations using the signal from the **LFO** and **Env** modules are controlled using the *LFO* and *Env* gain knobs respectively. The amplitude of the modulation is zero when the knob is centered. It is increased by moving it from its middle position clockwise or anti-clockwise. When turning it anti-clockwise, the phase of the modulating signal is inverted while it is preserved when moving it clockwise.

4.1.5 Synchronisation

The rate of the **Arpeggiator**, **LFO** and certain effect modules can be synchronized to the clock of a host sequencer when the program in used in plug-in mode. To do so, simply turn *on* their *Sync* switch. Synchronization values are adjusted with the *Sync Rate* parameter and range from 4 whole notes (16 quarter notes) to a thirty-second note (1/8 of a quarter note) where the duration of the whole note is determined by the host sequencer clock. The effect can also be synced to a triplet or dotted note. To adjust this parameter, click on the *Sync Rate* button and choose a rate value from the drop-down menu.

In standalone mode, when the *Sync* switch of an effect of module is switched *on*, the duration of a whole note is adjusted using the *Rate* control of the **Clock** module on the **Play** view.

4.2 The Edit View

The different synthesis modules appear in this view. There is a total of ten modules organized into four groups. One can switch from module to module by using the buttons located at the bottom of the *Edit* view.

4.2.1 The String Module

In a string instrument most of the sound we hear is radiated from the body of the instrument. The strings themselves radiate just a small amount of sound directly but it is their vibrations that are transmitted to the body of the instrument, through the bridge, where they can be radiated efficiently. It is also the strings that fix the pitch of the sound we hear depending on their effective lengths.

In a real string, the material of the string will affect how it vibrates. For example, a metal string will oscillate for a longer time than a nylon one; its sound will also be brighter. In the **String** module, this behavior is adjusted with the *Damping* and *Decay* knobs. The *Damping* knob is used to set the amount of high frequencies in the string vibration, this amount being increased as the knob is turned clockwise. The decay time of the vibrations is controlled with the help of the *Decay* knob and it is increased by turning the knob clockwise. Both of these parameters can be modulated with the pitch signal received from the keyboard.

In a first approximation, a string can be considered to be harmonic meaning that its partials are located at frequencies equal to multiples of its fundamental frequencies. Real strings, however, are more or less inharmonic depending mostly on the width of the string. This characteristic of strings is adjusted with the *Inharm* knob. When the *Inharm* knob is in its leftmost position, the string will be perfectly harmonic and turning the knob clockwise will increasingly detune the partials toward higher frequencies.

The *Release* knob is used to adjust the ratio between the decay time of the oscillation of the string when a note is depressed and when it is released. When the knob is in its leftmost position, both decay times are the same and equal to the decay time determined by the settings of the *Decay* knob. Turning this knob clockwise will decrease the decay time of the note when it is released while keeping the decay time when the key is depressed to its current setting. Note that this control constitutes an easy mean to reproduce the action of dampers on the string. When the **Damper** module is used and the *Release* knob is turned clockwise, the effect of the both damping mechanisms will add up.

Finally, the general level of the output signal from the **String** module is controlled with the *Level* knob. This parameter is proportional to the output signal from the body instrument sent into the distortion and the effects. It is therefore helpful to control the amount of distortion introduced by this modules and the different effects.

4.2.2 The Exciter Module

The **String** module can be played using different types of exciters in order to reproduce different types of instruments and playing techniques. The exciter is selected using the switches at the top of the module. The choices available are from left to right *Bow*, *Hammer 1*, *Hammer2* and *Plectrum*. These different types of exciters share the same front panel but note that the names of the parameters controlled by the different knobs vary for each exciter. We now review the different exciter types in more detail.

Plectrum

The *Plectrum* exciter, illustrated in Figure 9, is used to play instruments such as guitars, harpsichords or basses with a pick. The *Plectrum* can be viewed as an angled object placed under the

string and connected to a plate with the help of a spring. The purpose of the plectrum is to impose an initial displacement to the string before it is set into free vibration. As can be understood from figure 9, a vertical motion of the plate (which could be a hand holding the plectrum) will lift the string with the plectrum but will also result in a compression of the spring and an horizontal motion of the plectrum. The string will move with the plectrum until the protrusion (*Prot*) of the plectrum is equal to the compression of the spring and the string is released. The motion and behavior of the plectrum is controlled by adjusting the different geometrical and mechanical properties of the system.

The *Prot* knob is used to determine the protrusion of the plectrum with respect to the string while the stiffness and damping of the spring is controlled with the help of the *Stiff* and *Damp* knobs. The vertical velocity of the plectrum is adjusted with the *Velocity* knob. Note that the *Prot*, *Stiff*, and *Velocity* controls can be modulated with the pitch of the note played or the velocity signal from the keyboard.

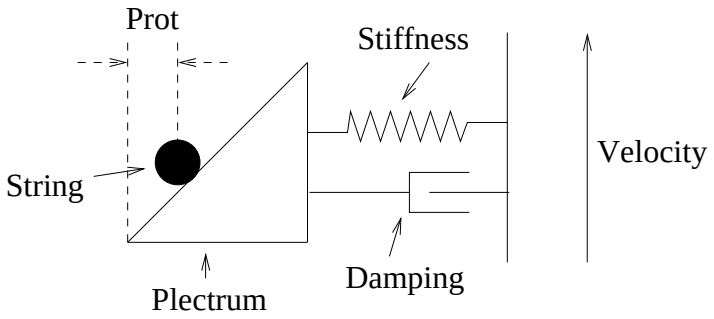


Figure 9: Functioning of the *Plectrum*

Hammer

The *Hammer* is used to play instruments such as the piano or other percussive instruments. With this exciter, the string is set into free vibration following a force impact with the hammer. The hammer can be used in two modes, *Hammer 1* and *Hammer 2*, as illustrated in Figure 10. In the *Hammer 1* mode, the hammer is located below the string and can only interact once with the string because of the action of gravity which brings it down after it has been raised to hit the string. In the *Hammer 2* mode, the hammer is located above the string and can bounce on the string after the initial impact.

The illustration in Figure 10 shows that the action of the hammer can be modeled by the motion of a head connected to a mass. The mass of the hammer is adjusted with the *Mass* knob while the stiffness of the head is controlled with the *Stiff* knob. The velocity of the hammer when it hits the string is set with the *velocity* knob. The motion of the hammer can further be characterized by a damping coefficient, adjusted with the

Damp knob, and controlling the absorption of the impact between the string and the hammer by the hammer. Note that this parameter is not related to the decay time of the string oscillation or the overall sound. On the contrary, the effect of this parameter may sometimes seem counter-intuitive even if it reproduces a physical property of the hammer. For example, increasing the damping of the hammer will make the compression of the spring linking the head to the mass harder and which will shorten the interaction between the hammer and the string but will also make it appear stronger resulting in a louder or longer sound.

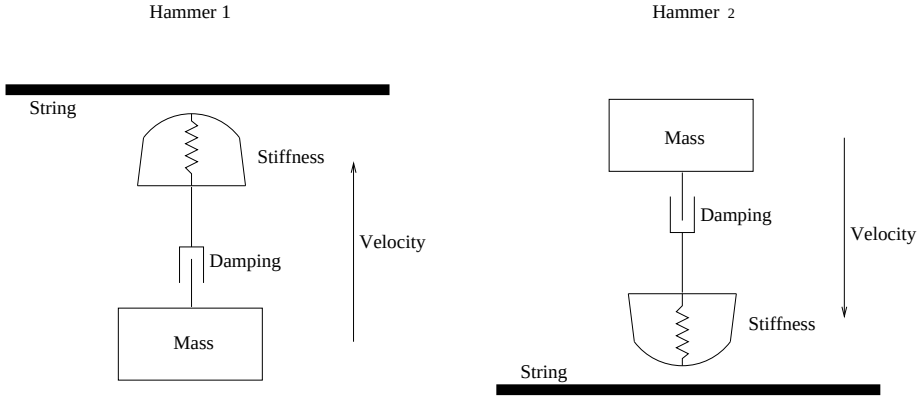


Figure 10: The two *Hammer* modes

Bow

The *Bow* exciter is used to play bowed instruments such as the violin, viola, or cello. The role of the bow is to set the string in self-sustained oscillation. Physically, oscillations of the string are maintained by a regular cycle of stick-slip movements. Due to friction forces between the string and the bow, the string sticks to the bow and follows its motion until the tension forces in the string, due to its own oscillating motion, break it free from the bow. The string is then in its slip phase and moves in the opposite direction to that of the bow. When the string motion changes direction once more, it sticks to the bow again, moving with the bow until it breaks free and repeats the cycle. Note that the frequency of this stick-slip motion is exactly the same as that of the string oscillation; or, in other words, the pitch of the note played.

The force with which the bow is applied on the string can be adjusted with the *Force* knob, the friction between the bow and the string is set with the *Friction* knob, and the velocity of the bow is controlled with the *Velocity* knob. Note that the *damp* knob is inactive when the *bow* exciter is selected. The tone and behavior of the instrument are the results of a complex relationship between these parameters but some general rules can however be followed. As the force applied by the bow on the string is increased, the tone becomes more scrubby. The friction between the bow and the string usually determines the length of the attack; the greater the friction, the faster the string can be set into motion. Finally, the velocity is related to the amplitude of the sound.

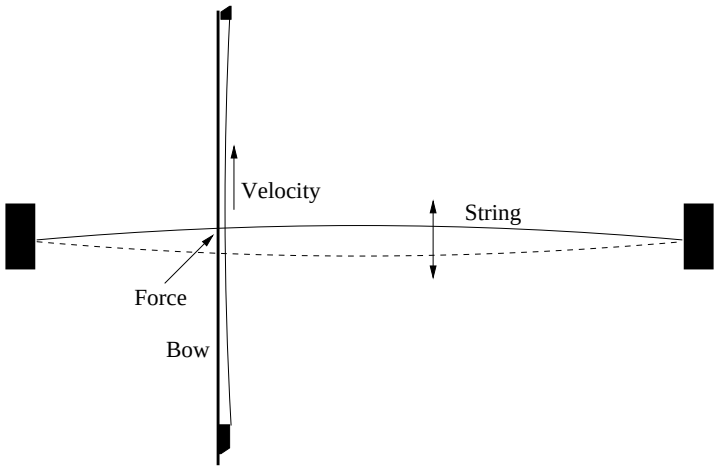


Figure 11: Excitation of a string by a bow

4.2.3 The Body Module

The role of the body or soundboard of a string instrument is to radiate the vibration energy from the strings. The body also adds a filtering effect to the vibration from the string which depends on its size and shape. In some instruments such as guitars, the body also includes an air cavity which boosts low frequencies.

The *Type* selectors located at the top of the module allow one to choose between different body geometries, each of them reproducing the spectral characteristics of the body of a specific type of instrument. For each type of body one can also determine its size with the help of the *Size* selector from tiny *T* to huge *H* (with small *S*, medium *M* and large *L* values in between). Basically, reducing the size of the **Body**, shifts its frequency response toward higher frequencies while increasing it, results in a shift toward lower frequencies. In addition to its shape and size, the material of the body also influences its radiation and filtering effects. This behavior is adjusted with the *Decay*, *Low Cut* and *High Cut* knobs. The decay time of the vibrations is controlled with the help of the *Decay* knob; it is increased by turning the knob clockwise. The *Low Cut* and *High Cut* knobs are used to set the amount of high and low frequencies respectively in the body vibration, this amount being increased as the knob is turned clockwise.

The *Mix* knob is used to adjust the ratio of direct signal from the *String* module and the signal filtered by the *Body* in the output signal of the **Body** module. In its leftmost position the output signal from the **Body** module will be that from the **String** module only while in its rightmost position, there is no direct signal from the **String** module. When this knob is in its center position, there is equal amounts of direct and filtered signal in the output signal of the **Body** module.

4.2.4 The Damper Module

The **Damper** module is used to attenuate rapidly the vibration of the string. In a piano or harpsichord, this role is played by felts while for the violin or the guitar, the performer's finger is used to damp the string vibrations. Basically, the damper can be viewed as a mass/spring system acting on the string as illustrated in Figure 12. The *Mass* and *Stiff* knobs are used to adjust these parameters, which affect how the damper interacts with the string. These physical parameters can be modulated with the pitch signal from the **Keyboard** and fine-tuned over the whole range of the instrument. The *Velocity* knob is used to adjust the velocity at which the damper is applied and released from the string. This parameter can also be modulated with the pitch signal from the **Keyboard** module. The last parameter of the **Damper** module is controlled with the *Damping* knob and refers to the ability of the damper to absorb energy from the string. Turning this knob clockwise will increase the damping exerted on the string by the damper.

Note that the **Damper** module responds to the sustain pedal signal via the Damper MIDI Control Change message (CC#64). In order for *String Studio VS* to respond to a sustain pedal, simply set your synth or MIDI controller to send its sustain pedal signal via this MIDI Control Change message.

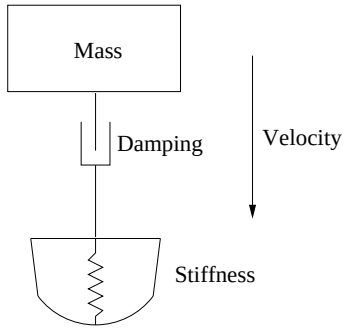


Figure 12: Functioning of the *Damper*

4.2.5 The Termination Module

This module is used to model the fret/finger/string interaction as illustrated in Figure 13. In a real instrument, this interaction is used to change the effective length of the string and thereby fix the pitch of the note played.

The physical parameters of the **Finger** can be varied with both the *Mass* and *Stifness* knobs which fix respectively mass applied on the string and the stiffness of the termination. Note that the *Mass* parameter can be modulated by both the pitch and velocity signal from the keyboard. The termination can further be characterized by the stiffness of the fret on which the string, pushed by the finger, is applied. This parameter is controlled by the *Stiff* knob under the **Fret** label.

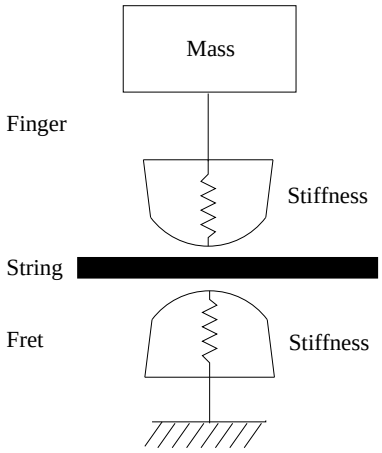


Figure 13: The finger/fret interaction

4.2.6 The Geometry Module

The **Geometry** module is used to set the location of the point of action of both the exciter and the damper on the string. These positions are adjusted with the *Position* knobs under the *Exciter* and *Damper* labels and can be set to any value between zero (the point of fixation of the string) and half the length of the string (value of 0.5).

When the *Abs* (absolute position) switch is *on*, the position of the exciter or the damper is fixed whatever the note played. This would be the case, for example, on a guitar when the player keeps the position of the pick fixed while varying the effective length of the string when changing notes. The actual position is determined with the setting of the *Position* knob applied to the length of a string corresponding to C3 (middle C). Note that when the note played is such that the string length corresponding to this note is shorter than this position, the exciter or the damper will follow the fixation point of the string.

When the *Abs* switch is in its *off* position, the location of the damper or the exciter is changed in order to always correspond to a certain fraction of the length of the string. This fraction of the

string length is that determined by the *Position* knob. This type of geometry is found in instruments such as the piano where hammers excite strings at about $1/7$ of their length.

Note that both the exciter and damper position can be modulated with the pitch signal or velocity signal received from the keyboard. The modulation will be relative to the value set by the exciter or damper *Position* knobs.

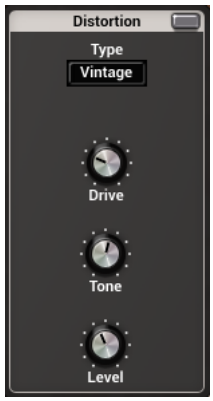
Pickup

The **Pickup** module reproduces the functioning of magnetic pickups such as found in electric guitars or electric pianos. This type of transducer is sensitive to the motion of a nearby metallic string. When a string vibrates near a pickup, the latter outputs an oscillating signal at the same frequency as that of the string and proportional to the string velocity.

The only parameter to adjust in the **Pickup** module is its position relative to the string which affects the waveform of its output.

Note that usually, the signal from a pickup is sent directly to an external device such as an amplifier. In other words, the body of the instrument does not play any role in the radiation of the sound. In *String Studio*, this behavior is obtained when the **Pickup** module is *on* and both the **Filter** and **Body** modules are switched *off*. When the **Pickup**, the **Filter** and **Body** modules are *on*, the output signal from the **Pickup** is filtered by the **Filter** and **Body** modules. Finally, when the **Pickup** module is switched *off*, the output signal from the **String** is sent directly to the **Filter** and **Body** module.

4.2.7 The Distortion module



The **Distortion** module implements a simple distortion effect, such as that found in electric guitar distortion pedals for example. Different distortion algorithms, ranging from *mellow* to *metal*, can be selected from the *Type* drop-down menu.

The *Drive* knob is a gain control used to adjust the level of the signal at the input of the **Distortion** module and hence the amount of saturation introduced in the signal. The color of the signal after the distortion algorithm has been applied can be adjusted using the *Tone* knob. In its leftmost position, high frequencies will be attenuated in the signal while in its rightmost position low frequencies will be filtered out from the signal. In its center position, the signal will be left unchanged. Note that this control can be set to its middle position by clicking on the small LED above the knob. Finally, the *Level* knob is used to control the amplitude of the signal at the output of the **Distortion** module.

4.2.8 The Filter Module

In order to expand the sonic possibilities of *String Studio*, a multi-mode filter has been inserted between the **String** and **Body** modules. This multi-mode filter includes a resonant low-pass, band-pass, high-pass, notch and a formant filter which can be selected using the *Type* drop-down menu. The order of the filter can be adjusted to 2 (-12 dB/oct for low-pass and high-pass and -6 db/oct for band-pass) or 4 (-24 dB/oct for low-pass and high-pass and -12 db/oct for band-pass) with the help of the *Order* drop-down menu. The resonance frequency of the filter is adjusted with the *Cutoff* knob while its Q-factor or resonance is controlled with the *Q* knob. When the formant filter is used, the *Q* knob is used to cycle between the vowels (a, e, i, o, u).

The cutoff frequency and resonance of the filters can be modulated with different modulation sources. The modulation sources include the keyboard pitch signal and the output of the envelope generator (**Env**) and **LFO** modules. Now let's have a closer look at the different filter types available.

Resonant Low-Pass Filter

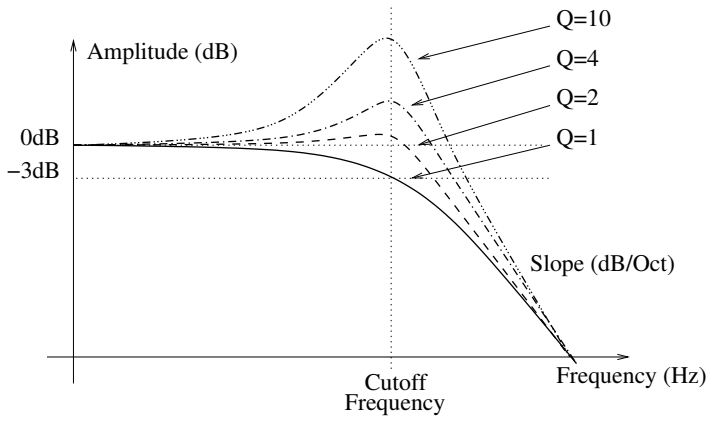


Figure 14: Frequency response of the low-pass filter.

A low-pass filter is used to remove the higher spectral components of the signal while leaving the lower components unchanged. The frequency at which attenuation begins to take effect is called the *cutoff* frequency. In a resonant filter, frequencies located around the cutoff frequency can also be emphasized by an amount called the *quality factor* or *Q-factor* of the filter as illustrated in Figure 14. The higher the Q-factor, the louder and sharper the response of the filter around the cut-off frequency. When the Q-factor is set to 1 (*Q* knob fully turned to the left), there is no emphasis around the cutoff frequency and the attenuation is -3dB at the cutoff frequency. The attenuation for frequencies located above the cut-off frequency depends on the order of the filter

which is determined by the *Order* menu, a slope of -12dB/Oct corresponding to a second order filter and a slope of -24dB/Oct to a fourth order filter.

Resonant High-Pass Filter

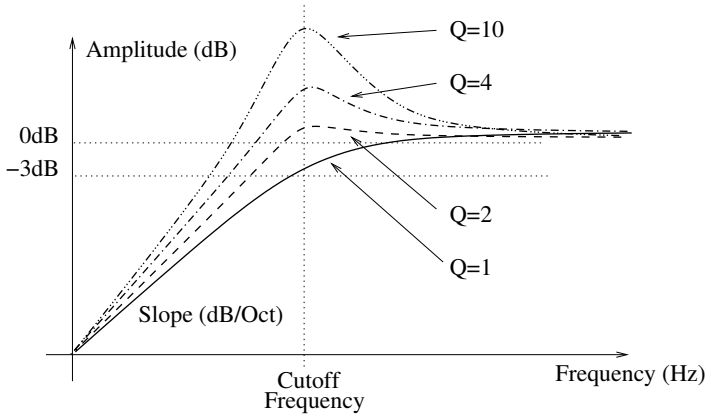


Figure 15: Frequency response of the high-pass filter.

The high-pass resonant filter works in exactly the opposite manner as the low-pass resonant filter by removing the frequency component of a signal located below the cutoff frequency while leaving those above the cutoff frequency unchanged. Similarly to the low-pass filter, the *Q-factor* controls the emphasis of frequencies located around the cut-off frequency.

Band-Pass Filter

The behavior of a band-pass filter is to let the frequencies in a band located around a center frequency and to attenuate the frequencies outside of this band as shown in Figure 16. The bandwidth of the band-pass filter is set with the *Q* knob while the center frequency is set with the *Cutoff* knob. The *Order* control sets the order of the filter. This parameter affects the slope of the roll-off on both sides of the center frequency. For a second order filter the slope is -6dB/Oct while for a fourth order filter it is -12dB/Oct.

Notch Filter

The notch filter does essentially the opposite of the band-pass filter. It attenuates the frequencies in a band located around the center frequency and leaves those outside of this band unchanged as shown in Figure 30. The *Cutoff* knob is used adjust the center frequency and the *Q* knob sets the bandwidth of the notch. Note that the center frequency is totally removed from the spectrum of the output signal of the filter.

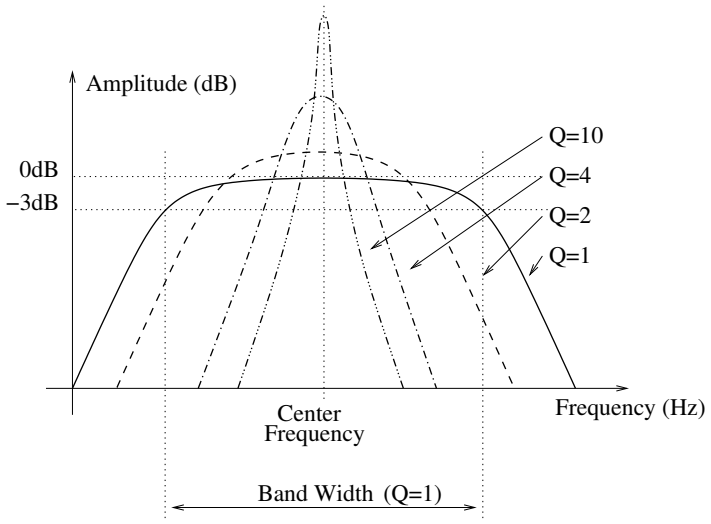


Figure 16: Frequency response of the band-pass filter.

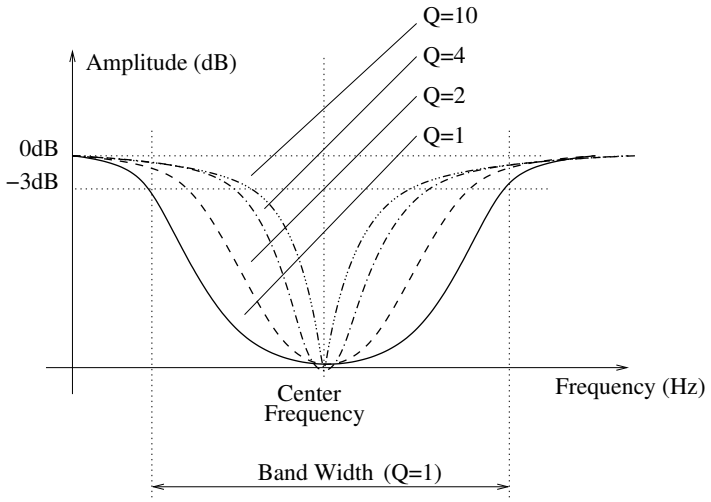


Figure 17: Frequency response of the notch filter.

Formant Filter

The formant filter reproduces the filtering effect of the vocal tract in the human voice. By changing the position of the tongue, the opening of the mouth and opening or closing the nasal cavities one can change the filter applied to the glottal signal and thus produce the different vowels. Measurements have shown that this filter can be modeled by three peaking EQ filters corresponding to the three main cavities of the vocal tract as shown in Figure 18 and also known as formants. By moving the parameters of these three filters (frequency, amplitude and Q-factor) one can cycle between

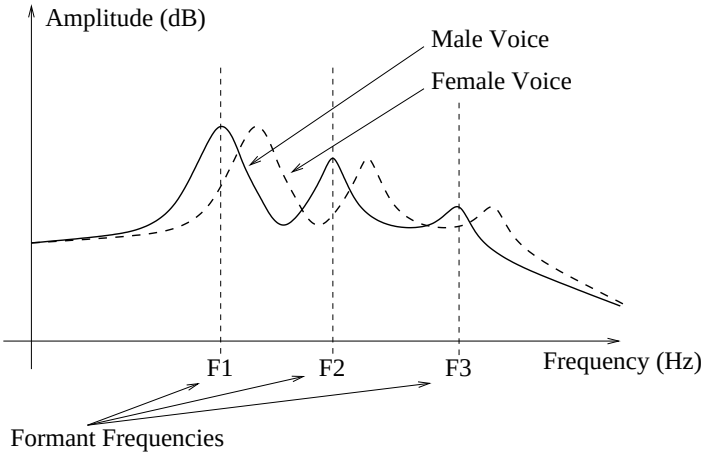


Figure 18: Frequency response of the formant filter.

all the vowels. The effect of the *Cutoff* knob on the formant filter is to offset all the formants by the same factor and it is used to switch between male voice (left position), female voice (center) and child (right position). The *Q* knob is used to cycle between vowels. Note that changing these parameters can be automated by using the different modulation signals.

4.2.9 The Envelope Module

The **Filter Env** envelope generator module is based on a standard ADSR (attack, decay, sustain, release) approach including velocity modulation.

The envelope module generates a four-segment envelope: attack, decay, sustain, release. The attack time is adjusted using the *A* knob. The attack time can also be modulated with the velocity signal received from the **Keyboard** in such a way that that the higher the velocity signal the shorter the attack time will be, the intensity of this effect being controlled using the modulation knob below of the *A* knob. When the knob is in its leftmost position, the attack is only determined by the value of the *A* knob, turning the knob clockwise will increase the influence of the velocity signal until the attack time is strictly determined by the inverse of the velocity signal when *V* reaches its maximum value. The decay time is set with the *D* knob. The sustain phase of the envelope generator lasts from the end of the decay phase until the key is released. When the *S* knob is fully turned to the left, the sustain level is zero and there is no sustain phase while fully turned to the right, the sustain level is at maximum and there is no decay phase. Note that the sustain phase can also be modulated with the velocity signal from the keyboard. Finally, when the key is released, the envelope generator toggles to the release phase and the envelope signal decreases from its sustain level to zero in a time set by the *R* knob.

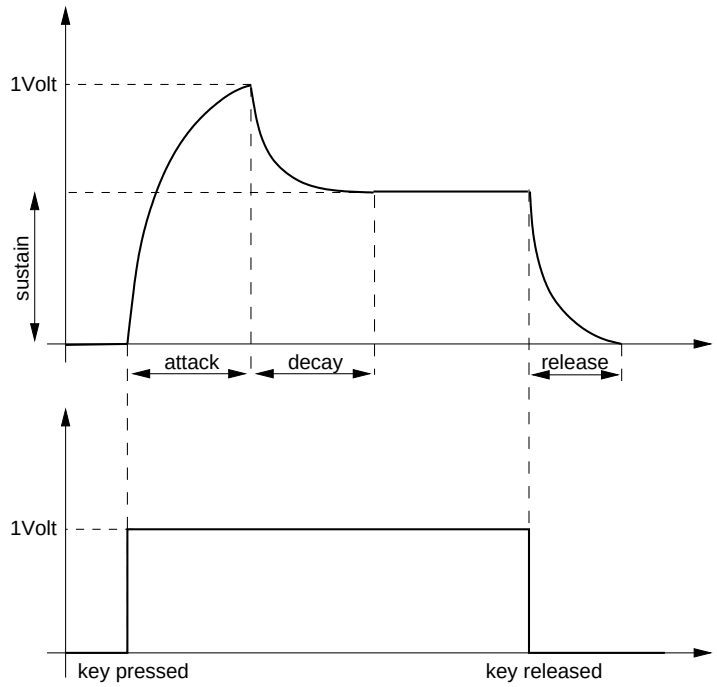


Figure 19: Response curve of an envelope generator

4.2.10 The LFO Module

The LFO module is used as a modulation source for the **Filter** module. On the **LFO** module, one can adjust the waveform, rate and fade-in behavior.

Wave Shape

The waveform of the **LFO** is selected with the *Shape* drop-down menu. The possible values are *Sine* for sinus, *Tri* for triangular, *Rect* for rectangular and *Rdm1* and *Rdm2* for the two random modes. When the *Shape* control is set to *Rdm1*, the LFO outputs random values at the rate determined by the *Sync Rate* control or the *Rate* knob. In this case, the output value from the LFO remains constant until a new random value is introduced. The *Rdm2* mode reacts almost like the preceding mode except that the **LFO** module ramps up or down between successive random values instead of switching instantly to the new value.

Rate

There are two ways to adjust the rate, or frequency, of the output of the **LFO** module. If the *Sync* switch is in its *off* position, the rate is fixed with the *Rate* knob. When the *Sync* switch is *on*, the frequency of the oscillator is fixed relative to the frequency (tempo) of the host sequencer or the master clock (see 4.4.1) in standalone mode. Sync values are adjusted using the *Sync Rate* control and range from 1/8 of a quarter note (a thirty-second note) to 16 quarter notes (4 whole notes). The **LFO** module can also be synced to a triplet (t) or a dotted note (d). Note that when the *Sync* control is depressed, the *Rate* knob has no effect.

Fade-In

One more feature of the **LFO** module is the possibility to add a fade-in effect to its output signal or in other words to set the amount of time necessary for the amplitude of the **LFO** signal to grow from zero to its maximum value. The duration of this fade-in can be adjusted within the range of 0 to 5 seconds, as determined by the *Fade* knob. Turning this knob fully to the left results in a value of 0 which is equivalent to removing the fade-in effect. The time at which the LFO signal is introduced can even be controlled by adding a delay to the fade in. This parameter can also be set to values varying between 0 and 5 seconds, as determined by the *Delay* knob. Note that this knob is effective even if the *Fade* value is adjusted to zero. In this case, the signal from the **LFO** module will simply be delayed.

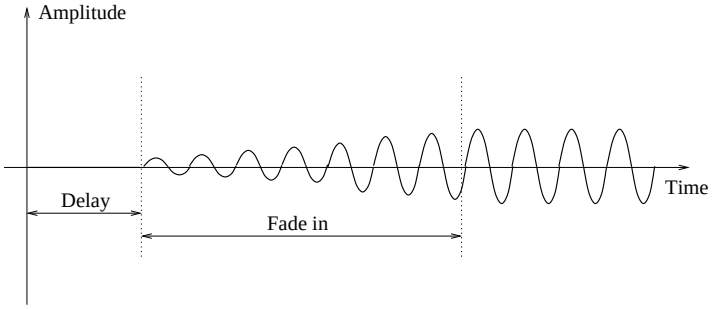


Figure 20: Fade in feature of the LFO.

4.3 The FX View

The **FX** view is displayed by clicking on the *FX* button in the utility section at the top of the interface and is based around a **Multi-effects** module.

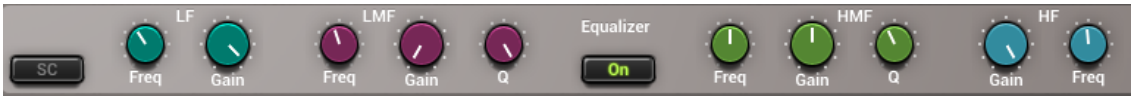
The **Multi-Effects** module allows one to process and shape the signal from the synthesizer before sending it to the output. This module comprises an **EQ** and a **Compressor** in series with

two configurable effect processors followed by a **Reverb** module. The configuration of the **EQ** and the **Compressor** module depends on the position of the *SC* and *Pre* butons of these modules as will be explained below. The two effect processors can be set to a different type by using the drop-down menu located at the center of each module for a wide range of possibilities. The effect list includes a **Delay**, **Distortion**, **Chorus**, **Flanger**, **Phaser**, **Wah Wah**, **Auto Wah** and a **Notch** filter. The order of the effect modules in the signal path can be changed by click-dragging the handles located on the left of the modules.

The **Multi-Effects** module is also visible from the **Play** view just below the utility section. This allows one to see rapidly which effects are selected for a given sound, turn the effects *on* or *off* and rapidly adjust the amount of each effect. The **Compressor**, **Equalizer** and **Reverb** can also be adjusted from this view.

4.3.1 EQ

The **EQ** module provides equalization over the low, mid, and high frequency bands. It is composed of a low shelf filter, two peak filters, and a high shelf filter in series, labelled **LF**, **LMF**, **HMF**, and **HF** respectively.



The functioning of the low shelf filter is depicted in Figure 21. The filter applies a gain factor to low frequency components located below a cutoff frequency while leaving those above unchanged. The cutoff frequency of this filter is adjusted using the *Freq* knob and can vary between 40 and 400 Hz. The *Gain* knob is used to adjust the gain factor applied to the signal in a $\pm$15dB range. In its center position there is no attenuation (0 dB). Turning it clockwise boosts the amplitude of low frequencies while turning it anti-clockwise reduces it.

The high frequency content of the signal is controlled with a high shelf filter that works in the opposite manner as the low shelf filter as illustrated in Figure 21. The filter applies a gain factor to components located above a cutoff frequency while leaving those below unchanged. The cutoff frequency of this filter, located above 1 kHz, is adjusted with the help of the *Freq* knob while the gain factor applied to the signal, in a $\pm$15dB range, is adjusted using *Gain* knob. In its center position there is no attenuation (0 dB). Turning it clockwise boosts the amplitude of high frequencies while turning it anti-clockwise reduces it.

The **EQ** module features two peak filters, labeled **LMF** and **HMF**, allowing to shape the signal in two frequency bands as illustrated in Figure 22. The filters apply a gain factor to frequency components in a band located around the cutoff frequency of the filters. This cutoff frequency is adjusted using the *Freq* knob and can vary between 100 Hz and 10 kHz. The gain factor applied a the cutoff frequency is controlled by the *Gain* knob and can vary in a $\pm$15 dB range. When in its center position there is no attenuation (0 dB). Turning it clockwise boosts the amplitude of frequencies located around the cutoff frequency while turning it anti-clockwise reduces it. The

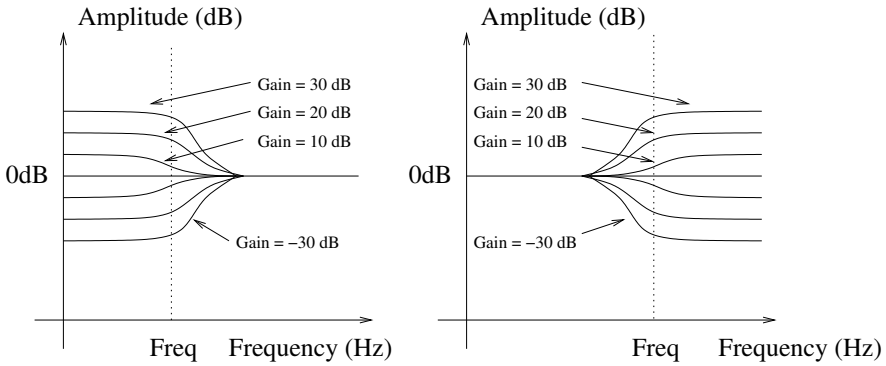


Figure 21: Low and high shelf filters.

Q knob is used to adjust the so-called quality factor of the filter which controls the width of the frequency band on which the filter is active. In its leftmost position, the frequency band is wide and it gets narrower as the knob is turned clockwise.

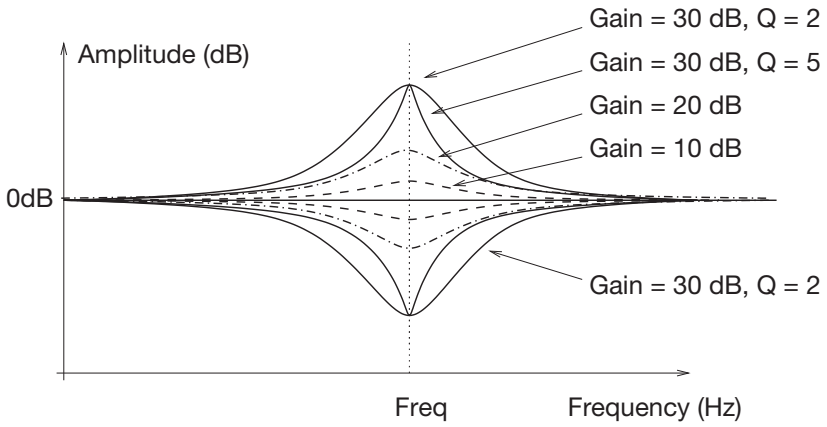


Figure 22: Peak filter.

The *SC* button (side-chain) is used to determine if the output from the **EQ** module is to be used as the control signal of the **Compressor** module as described in Section 4.3.2. Finally, note that all the gain knobs from this module can be accessed directly from the **Play** view.

4.3.2 Compressor

The **Compressor** module is used to automatically compress, in other words reduce, the dynamics of a signal. This module receives two input signals. The first one is the signal to be compressed while the second one is a control signal which triggers the compression process when it rises above a given level.

Tuning

The level at which the **Compressor** starts to enter into action is determined by the value of the *Threshold* parameter. This value is in dB and corresponds to the amplitude of the input signal as monitored by the first level meter of the module.

The amount of compression applied to the part of the signal exceeding the threshold value depends on the *Ratio* parameter which varies between value of 1:1 and 1:16. This parameter represents the ratio, in dB, between the portion of the output signal from the compressor above the threshold value and the portion of its input signal also exceeding the threshold value. As one might expect, increasing the ratio also increases the amount of compression applied to the signal. For example, a ratio of 1:5 means that if the input signal exceeds the threshold by 5 dB, the output signal will exceed the threshold by only 1 dB. Note that the *Ratio* parameter can also be adjusted from the **Play** view.

Two other controls affect the behavior of the **Compressor**. The *Attack* knob is used to set the time, in milliseconds, before the **Compressor** fully kicks in after the level of the input has exceeded the threshold value. A short value means that the compressor reaches the amount of compression as set by the *Ratio* knob rapidly. With a longer attack, this amount is reached more gradually. In other words, the attack time is a measure of the attack transient time of the compression effect. The **Release** parameter is similar and represents the amount of time taken by the **Compressor** to stop compressing once the amplitude of the input signal falls below the threshold value.

The **Makeup** knob is used to adjust the overall level at the output of the **Compressor** module and is used to compensate from an overall change in signal level due to the compression effect.

The location of the **Compressor** in the signal path depends on the setting of the *Pre* button. When this parameter is *on*, the **Compressor** is located at the output of the synthesizer, just before the **EQ** module. In this position, the input signal of the **Compressor** and its control signal are both the output signal from the synthesizer. When the **Pre** button is *off*, the **Compressor** is located after the **EQ** module. In this configuration, the control signal of the **Compressor** is then the output signal from the **EQ** module while the input signal to the compressor is determined by the position of the **SC** button. When it is *on*, the **Compressor** is in a *side-chain* configuration. The input signal of the **Compressor** is then the output signal from the synthesizer. When it is *off*, the input signal of the **Compressor** is the output signal from the **EQ** module.

Using the compressor in side chain configuration is useful when one wants to trigger the compressor using other criteria than the general level of the signal to be compressed. For example, a sound with a lot of bass would easily trigger the **Compressor** when playing low notes. In order to avoid that, the **EQ** module would be set to filter out low frequency components. This signal would then be used to control the **Compressor** while the input signal to the **Compressor** would still include these low frequency components.

The attenuation or gain reduction level meter, located in the middle of the module, indicates the amount of compression applied by the module. It is the difference between the input and output signals of the module before makeup gain is applied.

4.3.3 Delay

The **Delay** module consists in a stereo feedback loop with a variable delay in the feedback. It is used to produce an echo effect when the delay time is long (greater than 100 ms) or to color the sound when the delay time is short (smaller than 100 ms).

The *Delay knob* is used to adjust the amount of delay, in seconds, introduced by the effect. Turning this knob clockwise increases the delay. The *Feedback* parameter is a gain factor, varying in the range between 0 and 1, applied to the signal at the end of the delay lines. It controls the amount of signal that is re-injected in the feedback loop. In its leftmost position, the value of this parameter is 0 and no signal is re-introduced in the delay line which means that the signal is only delayed once. Turning the knob clockwise increases the amount of signal re-injected at the end of the feedback loop and therefore allows one to control the duration of the echo for a given delay time. In its rightmost position, the gain coefficient is equal to 1 which means that all the signal is re-injected into the feedback loop and that the echo will not stop. In addition to this gain factor, low pass filtering can also be applied to the signal re-injected into the feedback loop. The cutoff frequency of this filter is controlled using the *Cutoff* knob.

The *Pan* knob is used to balance the input signal between the left and right channels. In its leftmost position, signal is only fed into the left delay line and one hears clearly defined echo first from the left channel and then from the right channel and so on. In its rightmost position, the behavior is similar but with the first echo coming from the right channel. These two extreme position correspond to the standard ping pong effect but a a less extreme behavior can be obtained by choosing an intermediate position. In particular when the *Pan* knob is in its center position, an equal amount of signal is sent in both channels.

The output signal from the **Delay** module can include a mix of input signal (dry) and delayed signal (wet). The *Wet* and *Dry* knobs are used to adjust the amplitude of each component in the final output. The amplitude of each component is increased by turning the corresponding knob clockwise from no signal to an amplitude of +6dB. Note that the *Wet* parameter is also adjustable from the **Play** view.

4.3.4 Distortion

The **Multi-Effect** module includes three different types of distortion which are selected using the *Shape* selector knob. The *Warm Tube* effect applies a smooth symmetrical wave shaping to the

input signal resulting in the introduction of odd harmonics in the signal. The *Metal* distortion is similar to the *Warm Tube* effect but is slightly asymmetrical resulting in the introduction of even and odd harmonics in the signal. The *Solid State* distortion applies an aggressive symmetrical clipping to the signal thereby adding high frequency harmonics and resulting in a harsh sound.

The *Drive* control is a gain knob acting on the input signal. This parameter allows one to adjust the amount of distortion introduced in the signal by controlling how rapidly the signal reaches the non-linear portion of the distortion curve applied on the signal. In its leftmost position, the amplitude of the input signal is reduced by -6 dB; turning this knob clockwise allows one to increase its amplitude. Note that the *Drive* parameter is also adjustable from the **Play** view.

The *Tone* knob is used to adjust the color of the signal after the distortion algorithm has been applied. In its leftmost position, high frequencies are attenuated in the signal while in its rightmost position low frequencies are filtered out from the signal. In its center position, the signal is left unchanged.

The *Volume* knob is a gain knob acting on the amplitude of the distorted signal. Finally, the *Mix* knob allows one to control the amount of dry and wet (distorted) signal in the final output signal from the **Distortion** module. In its leftmost position, there is only dry signal in the output while in its rightmost position one only hears the distorted signal. In its center position, there is an equal amount of dry and wet signal in the output.

4.3.5 Chorus

The chorus effect is used to make a source sound like many similar sources played in unison. It simulates the slight variations in timing and pitch of different performers executing the same part. The effect is obtained by mixing the original signal with delayed version obtained from the output of delay lines as shown in Figure 23. In the case of a chorus effect, the length of the delay lines must be short in order for the delayed signals to blend with the original signal rather than be perceived as a distinct echo. The length of the delay line can be modulated introducing a slight perceived pitch shift between the voices.

Tuning

The amount of modulation of the length of the delay lines is adjusted using the *Depth* knob. In the left position, there is no modulation and the length of the delay lines remains constant. As the knob

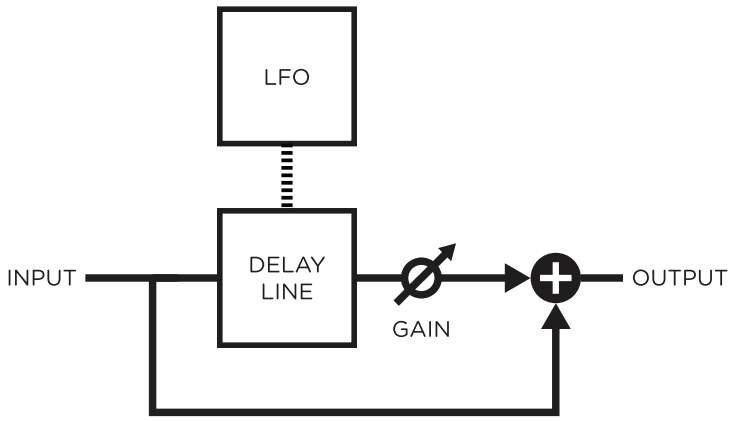


Figure 23: **Chorus** module.

is turned to the right, the length of the delay line starts to oscillate by an amount which increases as the knob is turned clockwise thereby increasing the amount by which the different voices are detuned. The frequency of the modulation is fixed with the *Rate* knob.

The *Fat* button is used to control the number of voices in the chorus effect. Switching this button *on* increases the number of voices. The *Spread* knob is used to adjust the amount of dispersion of the different voices in the stereo field. When in its leftmost position, there is an equal amount of left and right output signal on each channel. In other words the signal is the same on both channels. In its rightmost position, there is complete separation between the channels, the left output from the chorus is only sent to the left channel while the right output of the chorus is only sent to the right channel. Finally, the *Mix* knob allows one to mix the dry and wet signals. In its leftmost position, there is no output signal from the chorus and one only ears the dry input signals. In its rightmost position, one only ears the wet signal from the chorus module. In its center position, there is an equal amount of dry and wet signal in the output signal from the module.

Note that *String Studio VS* includes a second chorus algorithm called *vintage chorus* which sounds differently and has been kept for preset compatibility reasons.

4.3.6 Flanger

The **Flanger** module implements the effect known as *flanging* which colors the sound with a false pitch effect caused by the addition of a signal of varying delay to the original signal.

The algorithm implemented in this module is shown in Figure 24. The input signal is sent into a variable delay line. The output of this delay is then mixed with the dry signal and re-injected into the delay line with a feedback coefficient.

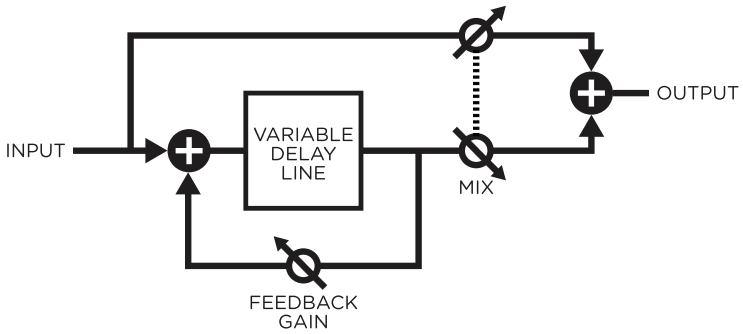


Figure 24: **Flanger** algorithm.

The effect of the **Flanger** module is to introduce rejection in the spectrum of the input signal at frequencies located at odd harmonic intervals of a fundamental frequency as shown in Figure 25. The location of the fundamental frequency $f0$ and the spacing between the valleys and peaks of the frequency response is determined by the length of the delay line ($f0$ = 1/(2delay)), the longer the delay, the lower is $f0$ and the smaller the spacing between the harmonics while decreasing the delay increases $f0$ and hence the distance between the harmonics.

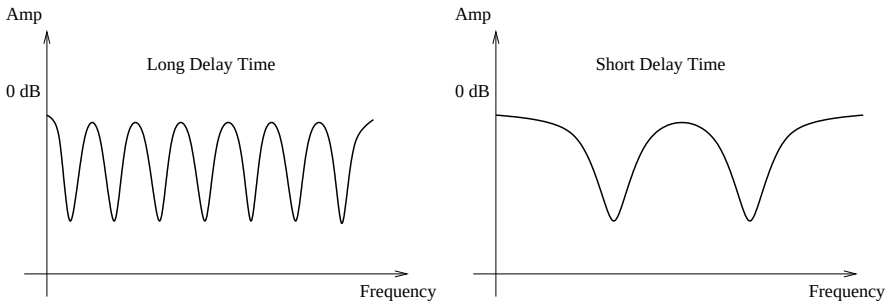


Figure 25: Frequency response of a **Flanger** module. Effect of the length of the delay line.

The amount of effect is determined by the ratio of wet and dry signal mixed together as shown in Figure 26. As the amount of wet signal sent to the output is increased, the amount of rejection increases. Finally, the shape of the frequency response of the **Flanger** module is also influenced by the amount of wet signal re-injected into the feedback loop as shown in Figure 27. Increasing the feedback enhances frequency components least affected by the delay line and located at even

harmonic intervals of the fundamental frequency. As the feedback is increased, these peaks become sharper resulting in an apparent change in the pitch of the signal.

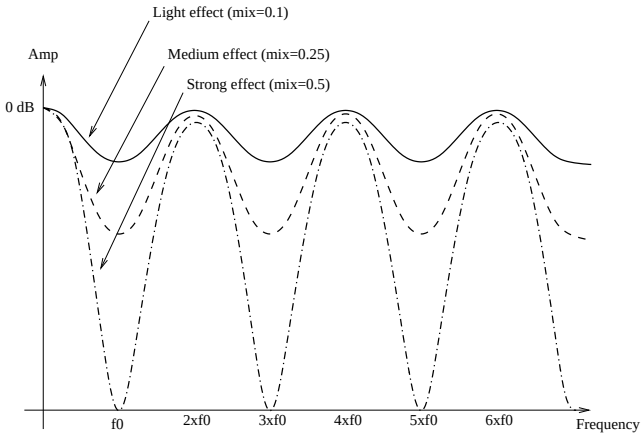


Figure 26: Effect of the mix between wet and dry signal on the frequency response of a **Flanger** module

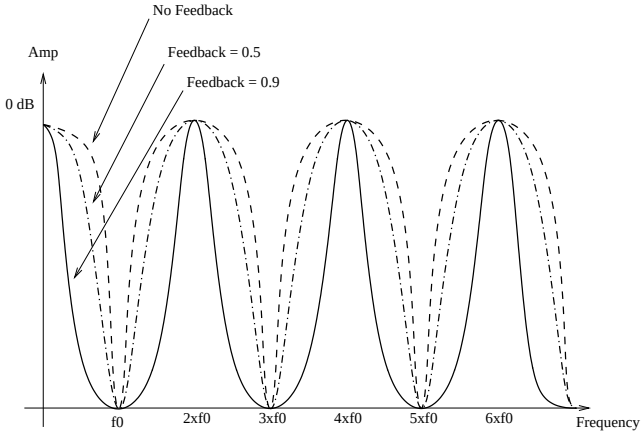


Figure 27: Effect of the amount of feedback on the frequency response of a **Flanger** module.

Tuning

The delay length, in milliseconds, is adjusted with the *Delay* knob. The length of this delay can be modulated by a certain amount depending on the adjustment of the *Depth* knob. In the left position, there is no modulation and the length of the delay line remains constant. As the knob is turned to the right, the length of the delay line starts to oscillate by an amount which increases as the knob is turned clockwise and at a frequency fixed with the *Rate* knob. The *Feedback* knob is a gain knob

used to fix the ratio of wet signal re-injected into the delay. Finally, the *Mix* knob determines the amount of dry and wet signal in the output signal from the module. When this knob is adjusted in its leftmost position, only dry signal is sent to the output, in its center position, there is an equal amount of dry and wet signal in the output signal while in its rightmost position, only wet signal is sent to the output. Note that the *Depth* parameter is also adjustable from the **Play** view.

4.3.7 Phaser

The **Phaser** module implements the effect known as *phasing* which colors a signal by removing frequency bands from its spectrum. The effect is obtained by changing the phase of the frequency components of a signal using an all-pass filter and adding this new signal to the original one.

The algorithm implemented in this module is shown in Figure 28. The input signal is sent into a variable all-pass filter. This wet signal is then mixed down with the original dry signal. A feedback line allows the resulting signal to be re-injected into the filter. The effect of the **Phaser** module is to introduce rejection in the spectrum of the input signal depending on the tuning of the filter.

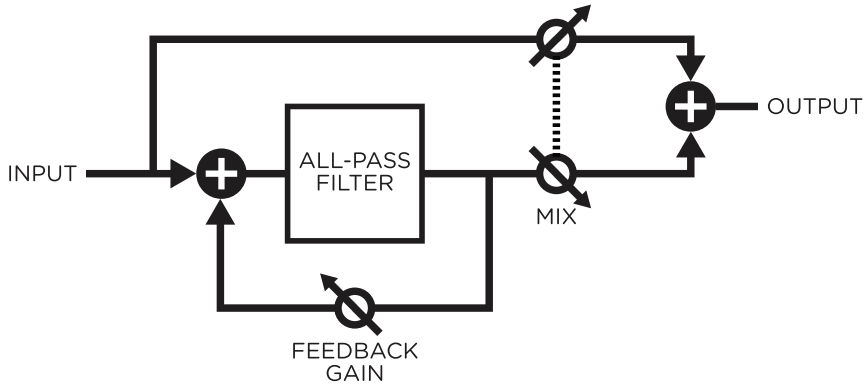


Figure 28: **Phaser** algorithm.

The all-pass filter modifies a signal by delaying its frequency components with a delay which increases with the frequency. This phase variations introduce a certain amount of cancellation when this wet signal is mixed down with the original dry signal as shown in Figure 29. The rejection is maximum when the phase delay is equal to 180 degrees and a given component is out of phase with

that of the original signal. The amount of effect is determined by the ratio of wet and dry signal mixed together as shown in Figure 29. As the amount of wet signal sent to the output is reduced, the amount of rejection increases. The shape of the frequency of the Phaser module is also influenced by the amount of wet signal re-injected into the feedback loop. Increasing the feedback enhances frequency components least affected by the all-pass filter. As the feedback is increased, these peaks become sharper. The functioning of the **Phaser** is very similar to that of the **Flanger** module. The filtering effect is different however, since the **Phaser** module only introduces rejection around a limited number of frequencies which, in addition, are not in an harmonic relationship.

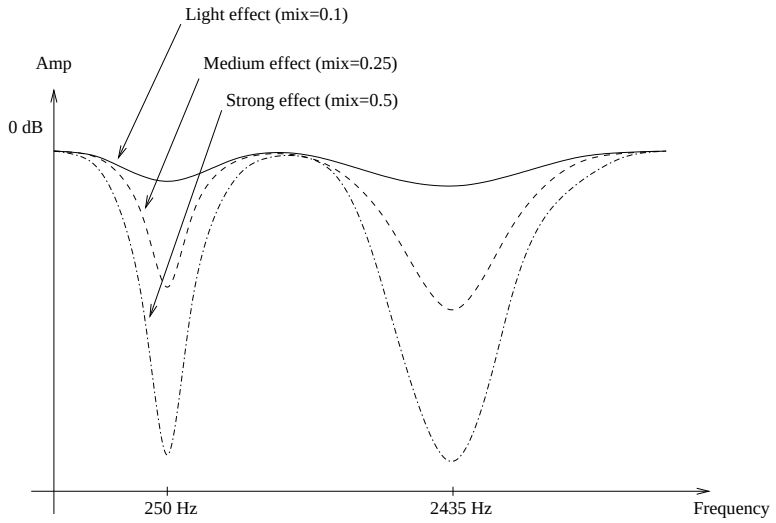


Figure 29: Frequency response of a **Phaser** module. Effect of the mix between wet and dry signal on the frequency response.

Tuning

The location of the first notch in the frequency response of the module is adjusted with the *Frequency* knob This frequency can be modulated by an amount controlled with the *Depth* knob. In its leftmost position, the location of the first notch is fixed but it starts to oscillate by an amount which increases as the *Depth* knob is turned clockwise. The frequency of the modulation is controlled using the the *Rate* knob. The *feedback* knob is used to fix the amount of wet signal re-injected into the delay. Finally, the *Mix* knob determines the amount of dry and wet signal sent to the output. When this knob is adjusted in the left position, only dry signal is sent to the output, in its center position, there is an equal amount of dry and wet signal in the output and in the right position, only wet signal is sent to the output.

4.3.8 Wah

The Multi-Effect module includes 2 different types of *Wah* effects: wah wah, and auto wah. These effects are used to enhance a frequency band around a varying center frequency using a bandpass filter. In the wah wah effect, the center frequency of the bandpass filter varies at a rate fixed by the user. In the case of the auto-wah, the variations of the center frequency is controlled by the amplitude envelope of the incoming signal.

The *Freq* knob is used to control the central frequency of the filter. Turning this knob clockwise increases the center frequency. In the case of the *Wah Wah* effect, the center frequency oscillates around the value fixed by the *Freq* knob while with the *Auto Wah* effect, the setting of the *Freq* parameter fixes the starting point value of the varying center frequency.

The *Depth* knob controls the excursion of the center frequency of the filter. In the case of the *Wah Wah* effect, this excursion is applied around the value fixed by the *Freq* knob while in *Auto Wah* effect the value of the center frequency increases from the value fixed by the *Freq* knob. Turning this knob clockwise increases the excursion of the center frequency. Note that the *Depth* parameter is also adjustable from the **Play** view.

Finally, the *Rate* knob controls the frequency or rate of the modulation of the center frequency of the filter. In the case of the *Wah Wah* effect, turning this knob clockwise increases the rate of the modulation. In the case of the *Auto Wah* filter, this knob is labeled *Speed* and controls the time constant of the envelope follower. Turning this knob clockwise decreases the time constant, or in other words the reaction time, of the envelope follower.

4.3.9 Notch Filter

The *Notch Filter* does essentially the opposite of a band-pass filter. It attenuates the frequencies in a band located around the center frequency and leaves those outside of this band unchanged as shown in Figure 30. As was the case for the *Wah Wah* effect, the filter can be modulated.

The *Freq* knobs is used to control the central frequency of the filter. Turning this knob clockwise increases the center frequency. The *Depth* knob controls the excursion of the center frequency of the filter around its center frequency. Turning this knob clockwise increases the excursion of the center frequency. Finally, the *Rate* knob controls the frequency or rate of the modulation of the center frequency of the filter. Turning this knob clockwise increases the rate of the modulation. Note that the *Depth* parameter is also adjustable from the **Play** view.

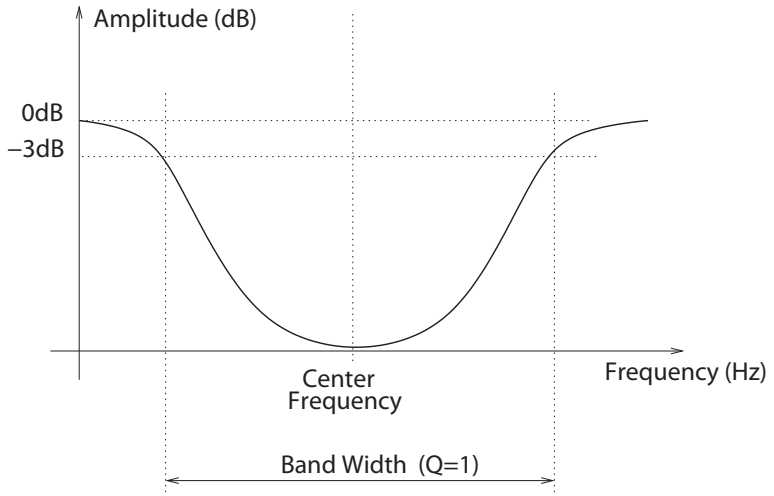
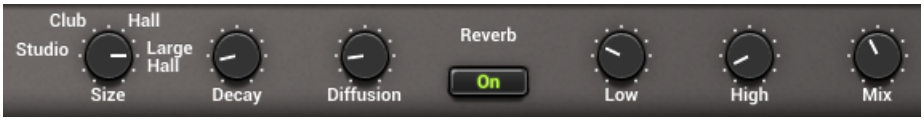


Figure 30: Frequency response of a notch filter.

4.3.10 Reverb

The *Reverb* effect is used to recreate the effect of reflections of sound on the walls of a room or hall. These reflections add space to the sound and make it warmer, deeper, as well as more realistic since we always listen to instruments in a room and thus with a room effect.



Impulse Response of a Room

The best way to evaluate the response of a room is to clap hands and to listen to the resulting sound. Figure 31 shows the amplitude of the impulse response of a room versus time. The first part of the response is the clap itself, the direct sound, while the remaining of the response is the effect of the room which can itself be divided in two parts. Following the direct sound, one can observe a certain amount of echoes which gradually become closer and closer until they can not be distinguished anymore and can be assimilated to an exponentially decaying signal. The first part of the room response is called the early reflexion while the second is called the late reverberation. The total duration of the room response is called the reverberation time (RT).

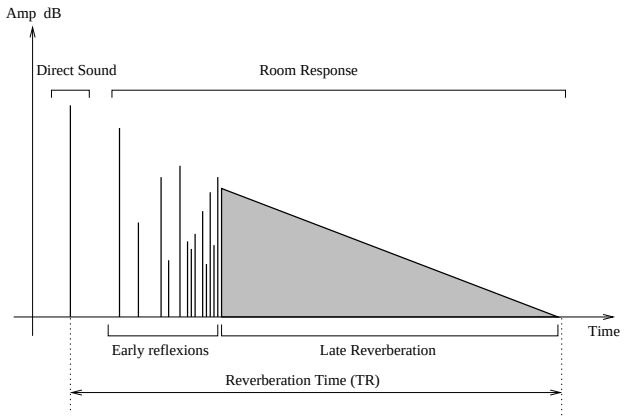


Figure 31: Impulse response of a room.

Adjusting the room effect

The size of a room strongly affects the reverberation effect. The *Size* selector is used to choose between the *Studio*, *Club*, *Hall* and *Large Hall* settings each reproducing spaces of different volumes from smaller to larger.

The duration of the reverberation time depends on both the size of the room and the absorption of the walls, which is controlled with the *Decay* knob. In a real room the reverberation time is not constant over the whole frequency range. As the walls are often more absorbent in the very low and in the high frequencies the reverberation time is shorter for these frequencies. These parameters are adjusted with the *Low* and *High* knobs respectively.

Another parameter which affects the response of a room is its geometry; the more complex the geometry of a room, the more reflexion are observed per unit of time. This quantity is known as the time density and can be set trough the *Diffusion* knob. In a concert hall, the time density is supposed to be quite high in order not to hear separate echoes which are characteristic of poor sounding rooms.

The last parameter which affects our listening experience in a room, is the distance between the sound source and the listener. While the room response is quite constant regardless of the position of the source and the listener, the direct sound (the sound which comes directly from the source) depends strongly on the position of the listener. The farther we are from the sound source the quieter is the direct sound relatively to the room response. The ratio between the direct sound and the room response is adjusted with the *Mix* knob which in other words is used to adjust the perceived distance between the source and the listener. In its leftmost position, only the direct sound is heard while when fully turned to the right, one only hears the room response. Note that the *Mix* parameter is also adjustable from the **Play** view.

4.4 The Play View

The **Play** view is where the main performance oriented modules are located. Key parameters from the **Edit** and **FX** view are also included for quick access. This view is loaded when starting the instrument and can be accessed from another view by clicking on the *Play* button on the top part of the interface.

The middle section of this view allows one to switch *on* and *off* the **EQ**, **Compressor** and **Reverb** as well as the active effect modules. Key effect parameters are also adjustable as presented in the description of the different effect modules in section 4.3

4.4.1 The Clock Module

This module is used to control the tempo of the different effects of the FX section as well as that of the **LFO** and **Arpeggiator** modules when their respective *sync* button is switched *on*. When *String Studio VS* is launched in standalone mode the clock tempo, in bpm, is set by using the *Rate* knob. The tempo can also be adjusted by clicking at the desired tempo on the *Tap Tempo* pad of the module. Once the new tempo is detected, the value of the *Rate* knob is automatically adjusted.

When using *String Studio VS* in plugin mode, the *Tap Tempo* pad is replaced by a *Sync To Host* switch. In its *on* position, the rate is synchronized with that of the host sequencer. When switched *off*, the tempo is determined by the value of the *Rate* knob.

4.4.2 The Key Module

The **Key** module controls how the synthesizer voices respond to the events coming from an external MIDI keyboard or from a MIDI sequencer.

The keyboard can be monophonic, allowing one to play only one note at a time, or polyphonic, allowing one the play chords. This behavior is adjusted with the *Mode* drop-down menu. The *Priority* control, located below the *Mode* control, sets the behavior of the keyboard when several notes are depressed at the same time in monophonic mode or when the maximum number of polyphonic voices has been reached in polyphonic mode. In monophonic mode, the *Priority* determines which of the lower, last, or higher note has precedence when several notes are played. In polyphonic mode, this control determines which of the lowest, highest, or oldest note is muted in order to replace it with the newest note played once the maximum of polyphonic voices has been reached. Note that since this parameter determines the note priority, the stolen note will be the opposite of what appears in the control display. The

Error knob adds a random error to the pitch of each note played; as it is turned clockwise, the error increases. This makes all the notes play slightly out of tune.

Note that the sustain pedal (MIDI controller number 64) is always active on the **Key** module.

4.4.3 Unison

The unison mode allows one to stack voices, in other words, play two or four voices for each note played on the keyboard. This mode creates the impression that several instruments are playing the same note together, adding depth to the sound.

Each voice can be slightly detuned relatively to the others by using the *Detune* knob. Turning this knob clockwise increases the amplitude of the error. Furthermore, voices can be desynchronized by adding a small time lag between their triggering with the *Delay* knob. There is no delay when the knob is in its leftmost position and it increases (units in ms) as it is turned clockwise.

4.4.4 The Glide Module

The **Glide** module is used to make the pitch slide between notes rather than changing immediately from note to note. The *Time* knob sets the amount of time necessary for the pitch to slide over one octave. The *Mode* drop-down menu enables one to choose between the *Constant* or *Proportional* mode. When in *Constant* mode, the time necessary for the pitch to slide from one note to another is always the same regardless of the interval between the notes. When set to proportional, the slide time becomes proportional to the width of the interval between the two notes.

Clicking on the *Legato* button switches the module into legato mode and the sliding between two notes then only occurs if the second note is played before the first one is released. When a note is played staccato, or in other words if a key is released before the next one is depressed, there is no glide effect. Note that even though the glide effect is available when the **Key** module is in polyphonic mode, it is mostly dedicated to monophonic playing. In polyphonic mode, the sames rules apply to individual voices and the overall result is less predictable.

4.4.5 The Vibrato Module

The vibrato effect is equivalent to a periodic low frequency pitch modulation. This effect is generally obtained by using an LFO to modulate the pitch signal of an oscillator. In *String Studio VS*, a dedicated module is provided for this effect. The vibrato module is hard wired and affects the pitch of both oscillators.

The *Rate* knob sets the frequency of the vibrato effect from 0.3 Hz to 10 Hz. The *Amount* knob sets the depth of the effect, or in other words the amplitude of the frequency variations. In its leftmost position, there is no vibrato and turning the knob clockwise increases the amount of pitch variation. The *MW* gain knob is used to determine the effect of the keyboard modulation wheel on the depth of the vibrato. When this knob is fully turned to the left, the modulation wheel has no effect but as it is turned clockwise the depth of the vibrato increases when the modulation wheel is used. The increase is always relative to the position of the *Amount* knob and becomes greater as the *MW* knob is turned clockwise.

The vibrato can be adjusted not to start at the beginning of a note but with a little lag. This lag, in seconds, is set by the *Delay* knob. The *Fade* knob allows you to set the amount of time taken by the amplitude of the vibrato effect to grow from zero to the amount set by the *Amount* knob.

4.4.6 The Arpeggiator Module

The **Arpeggiator** module allows one to play sequentially all the notes that are played on the keyboard. In other words, arpeggios are played rather than chords. The modules allows one to produce a wide range of arpeggios and rhythmic patterns and to sync the effects to the tempo of an external sequencer.

Arpeggio Patterns

The arpeggio pattern is set by the combination of the value of the *Range*, *Span* and *Order* controls. The *Range* control is used to select the number of octaves across which the pattern is repeated. When the range is set to 0, there is no transposition and only the notes currently depressed are played. If set to a value between 1 and 4 (its maximum value), the notes played are transposed and played sequentially, over a range of one or more octaves depending on the value of the *Range* parameter. The direction of the transposition is set with the *Span* drop-down menu. This parameter can be adjusted to *Low* for downwards transposition, to *High* for upwards transposition or *wide* for transposing both upwards and downwards. Finally, the *Order* control sets the order in which the notes are played, therefore determining the arpeggio pattern. When set to *Forward*, the notes are played from the lowest to the highest. When set to *Backward* the notes are played from the highest to the lowest. In the two last modes, *Rock and Roll exclusive* and *Rock and Roll inclusive*, the notes are played forward from the lowest to the highest and then backward from the highest down to the lowest. When using the *RnR exclusive* mode, the highest and the lowest notes are not repeated when switching direction but in RnR inclusive mode these notes are repeated. Finally, in *Chord* mode, all the notes are played at once.

Rhythmic Patterns

Rhythmic patterns can be added to the arpeggio pattern by using the 16-step *Pattern* display. Notes are played as the step display is scanned and the corresponding step is selected (red button *on*). Notes are played regularly when all the steps of the display are turned *on* and rhythmic patterns are created by selecting only certain steps. The arrow button below each step is used to fix looping points from which the rhythmic pattern starts being played again from the beginning.

Rate and Synchronization

The rate at which the arpeggiator pattern is scanned is set by the *Rate* knob of the **Arpeggiator** module or can be synced to the master clock of the *Clock* module. The *Rate* knob is only effective when the *Sync* control is set to *off*. When the *Sync* control is *on*, the rate (tempo) is fixed by the master **Clock** module (see 4.4.1) in standalone mode or the host sequencer in plugin mode. The rhythmic value of each step is set using the *Steps* parameter. Values can range between a quarter note and a thirty-second note with binary and ternary beat division options. One can then fix the metric of the pattern by setting the loop point of the step display appropriately.

Latch mode

The **Arpeggiator** module is toggled in latch mode by clicking the *Latch* button to its *on* position. In this mode, the **Arpeggiator** keeps playing its pattern when the notes on the keyboard are released and until a new chord is played.

4.4.7 Pitch Wheel

The MIDI pitch wheel allows one to vary the pitch of the note played. The pitch wheel can be moved with the mouse but it is also automatically connected to the pitch wheel signal received from your MIDI keyboard.

The range of the pitch bend is 2 semi-tones up or down by default but can be changed. To adjust the range of the pitch bend, open the MIDI configuration window by clicking on the **MIDI** button located just below the MIDI led in the top part of the interface and use the **Pitch Bend Range** drop-down menu to select the range in semi-tones.

4.4.8 Modulation Wheel

The modulation wheel is linked to the *Amount* parameter of the **Vibrato** module. It can be activated on screen or from the modulation wheel of your MIDI controller (MIDI controller number 1). The *MW* gain knob of the **Vibrato** module is used to control the sensitivity of the vibrato amplitude to the modulation wheel. Note that other parameters can be linked to the modulation wheel using MIDI links as explained in Section 6.2.2.

4.4.9 Ribbon

The lower part of this view includes a ribbon controller. The ribbon covers seven octaves and notes are played when clicking on the ribbon. The ribbon is useful to test sounds when no MIDI keyboard is connected to your computer.

5 Utility Section

The utility section is located at the top of the *String Studio VS* interface and it includes important parameters and monitoring tools. For information on *Banks* and *Programs* please refer to Chapter 3.

5.1 The MIDI LED

The MIDI LED is located on the left of the level-meter. The LED blinks when the synthesizer receives MIDI signal. If the application is not receiving MIDI signal, make sure that the host sequencer is sending MIDI to *String Studio VS*. If you are running in standalone mode, make sure that the MIDI controller you wish to use is well connected to your computer and that it is selected as explained in Section 6.

Clicking on the *MIDI* button just below the MIDI led opens the MIDI configuration pop-up window from which one can adjust parameters such as the pitch bend range, enable bank and program changes and manage the MIDI map used by *String Studio VS* as explained in Chapter 6

5.2 Tuning

The *Tune* control, located to the right of the MIDI LED, is used to transpose the frequency of the keyboard. This control is composed of two numbers separated by a dot. The first number indicates a value in semi-tones while the second one indicates a value in cents (one hundredth of a semi-tone). The amount of transposition can be adjusted by click-dragging upward or downward on the semi-tone and cent controls. Double clicking on these controls brings back their value to zero. When the value of the *Tune* parameters is set to 0.00, the frequency of notes are calculated relative to A4 440Hz.

An interesting feature of *String Studio VS* is that it can be tuned using different temperaments using Scala micro-tuning files. Temperament files are loaded by clicking on the *Tune* button which opens the Tuning pop-up window and displays the list of available tuning temperament files available. By default, *String Studio VS* is set to Equal Temperament but other temperament files can be added to the list by copying them to the following folders:

On Mac OS: /Users/[user name]/Library/Application Support/Applied Acoustics Systems/Scala Tunings/ On Windows:

These folders can be displayed directly from String Studio VS by clicking on the *Show Tuning Files* button at the bottom of the *Tuning* pop-up window.

Selecting a Scala file in the list automatically triggers the loading of the corresponding temperament. The reference note that will be used as the base note for the scale described in the Scala file can be set using the Reference Note control appearing at the bottom of the *Tuning* window. The frequency of this reference note is calculated relative to the settings of the *Tune* control.

5.3 Polyphony

The *Voices* control located below the *Tune* button allows one to adjust the number of polyphony voices used by *String Studio VS*. The number of voices is adjusted by clicking on the control and selecting the desired number of voices. In general, a higher number of voices is desirable but keep in mind that the CPU load is proportional to the number of voices used.

5.4 History and Compare

The *History* control allows one to go back through all the modifications that were made to programs since the application was started. In order to travel back and forth in time, use the left and right-pointing arrows respectively. The application will switch between different program states and indicate the time at which they were modified.

The *Compare* button, located above the *Program* display, is used to switch between **Edit** and **Compare** mode. This button is visible only once a modification is applied to a given program. It allows one to revert to the original version of a program in order to compare it with the current version. When in **Compare** mode, edition is blocked and it is therefore not possible to modify any parameter. The **Compare** mode must then be switched *off* by clicking on the *Compare* button in order to resume edition.

5.5 Volume

The *Volume* knob is the master volume of the application. It is used to adjust the overall level of the output signal from the synthesizer. General level is increased by turning the knob clockwise.

5.6 Level Meter

The level meter allows one to monitor peak and RMS (root means square) level of the left (L) and right (R) output channels from the synthesizer. As a limiter is located at the output of *String Studio VS*, it is important to make sure that the amplitude of the signal remains within values that ensure that no distortion is introduced in the signal at the output.

The 0 dB mark on the level meter has been adjusted to correspond to -20 dBFS (full scale). This means that at that level, the signal is -20 dB below the maximum allowed value. This 0 dB level mark should typically correspond to playing at mezzo forte (moderately loud) level. This ensures

a headroom of 20 dB which should be more than enough to cover the dynamics of most playing situations and therefore guarantee that no additional distortion is added in the output signal.

A peak value mark allows one to follow the maximum level values reached by the output signal. The limiter is triggered when this mark enters the red zone of the level meter (17 dB) and it remains active while the side vertical bars at the top of the lever meter are switched *on*.

5.7 The About Box

The **About** box is open by clicking on the chevrons located at the very top of the interface or on the product or company logo. The box is closed by clicking again on the chevrons or outside the box. Useful information is displayed in this box such as the program's version number, the serial number that was used for the authorization and the the email address that was used for registration. The box also includes a link to the pdf version of this manual.

6 Audio and MIDI Settings

This chapter explains how to select and configure Audio and MIDI devices used by *String Studio VS*. Audio and MIDI configuration tools are accessed by clicking on the *Audio Setup* button located in the lower left corner of the *String Studio VS* interface and the *MIDI* button located just below the MIDI led in upper part of the interface.

Note that in plug-in mode the audio and MIDI inputs, sampling rate, and buffer size are set by the host sequencer.

6.1 Audio Configuration

6.1.1 Selecting an Audio Device

Audio configuration tools are available by clicking on the *Audio Setup* button located in the lower left corner of the *String Studio VS* interface. The **Audio Setup** dialog first allows you to select an audio output device from those available on your computer. Multi-channel interfaces will have their outputs listed as stereo pairs.

On Windows, the audio output list is organized by driver type. The device type is first selected from the **Audio Device Type** drop-down list. If you have ASIO drivers available, these should be selected for optimum performance. The *Configure Audio Device* button allows you to open the manufacturer's setup program for your audio interface when available.

Once the audio input has been selected, you can then select a sampling rate and a buffer size from those offered by your audio interface.

6.1.2 Latency

The latency is the time delay between the moment you send a control signal to your computer (for example when you hit a key on your MIDI keyboard) and the moment when you hear the effect. Roughly, the latency is equal to the duration of the buffers used by the application and the sound card to play audio and MIDI. To calculate the total time required to play a buffer, just divide the number of samples per buffer by the sampling frequency. For example, 256 samples played at 48 kHz represent a time of 5.3 ms. Doubling the number of samples and keeping the sampling frequency constant doubles this time while changing the sampling frequency to 96 kHz and keeping the buffer size constant reduces the latency to 2.7 ms.

It is of course desirable to have as little latency as possible. *String Studio VS* however requires a certain amount of time to be able to calculate sound samples in a continuous manner. This time depends on the power of your computer, the preset played, the sampling rate, and the number of voices of polyphony used. Note that it literally takes twice as much CPU power to process audio at a sampling rate of 96 kHz as it would to process the same data at 48 kHz, simply because you need to calculate twice as many samples in the same amount of time.

Depending on your machine you should choose, for a given sampling frequency, the smallest buffer size that allows you to keep real-time for a reasonable number of voices of polyphony.

6.2 MIDI Configuration

6.2.1 Selecting a MIDI Device

The list of available MIDI inputs appears at the bottom of the **Audio Setup** dialog. Click on the *Audio Setup* button located in the lower left corner of the *String Studio VS* interface and then click on the checkbox corresponding to any of the inputs you wish to use.

6.2.2 Creating MIDI Links

Every control on the *String Studio VS* interface can be manipulated by an external MIDI controller through MIDI control change assignments. In most cases this is much more convenient than using the mouse, especially if you want to move many controllers at once. For example, you can map the motion of a knob on the interface to a real knob on a knob box or to the modulation wheel from your keyboard. As you use the specified MIDI controllers, the controls move on the *String Studio VS* interface just as if you had used the mouse.

In order to assign a MIDI link to a controller:

- On the *String Studio VS* interface, right-click/Control-click on a control (knob, button) and select the **MIDI Learn** command.
- Move a knob or slider on your MIDI controller (this can be a keyboard, a knob box, or any device that sends MIDI). This links the control of *String Studio VS* to the MIDI controller you just moved.

To deactivate a MIDI link, simply righ-click/Control-click on the corresponding control on the *String Studio VS* interface and select the **MIDI Forget** command.

6.2.3 Creating a default MIDI Map

It is possible to define a a set of MIDI links, called a MIDI map, that will be loaded automatically when *String Studio VS* is launched. Once you have defined a set of MIDI links that you wish to save, click on *MIDI* button to open the *MIDI* configuration window and click on the **Save Current as Default** button.

If you make changes to MIDI links after opening the program and wish to revert to the default MIDI map click on *MIDI* button to open the *MIDI* configuration window and click on the **Load Default** button.

If you wish to deactivate all the MIDI links at once open the *MIDI* configuration window and click on the **Clear MIDI Map** button.

6.2.4 MIDI Program Changes

String Studio VS responds to MIDI program changes. When a program change is received, the current program is changed to the program having the same number as that of the program change message in the currently loaded bank.

If you do not wish *String Studio VS* to respond to MIDI program changes, open the *MIDI* configuration window by clicking on the *MIDI* button and uncheck the **Enable Program Changes** option.

6.2.5 MIDI Bank Changes

In general, MIDI bank numbers are coded using two signals: the MSB (most significant byte) and LSB (least significant byte) transmitted using MIDI CC (continuous controller) number 0 and 32 respectively. The way these signals are used differs with different manufacturers.

In the case of *String Studio VS*, the value of the MSB signal is expected to be zero while the value of the LSB signal represents the bank number. Banks are therefore numbered from 0 to 127 with this number corresponding to the position of a bank within the list of banks as displayed by the Bank manager (see Section 3.3). For example, an LSB value of 0 corresponds to the first bank in the bank list while an LSB value of 10 corresponds to the eleventh bank in the list. Note that a bank change only becomes effective after the reception of a new MIDI program change signal.

If you do not wish *String Studio VS* to respond to MIDI bank changes, open the *MIDI* configuration window by clicking on the *MIDI* button and uncheck the **Enable Bank Changes** option.

6.2.6 Pitch bend

The MIDI pitch wheel allows one to vary the pitch of *String Studio VS*. The pitch wheel can be moved with the mouse but it is also automatically connected to the pitch wheel signal received from your MIDI keyboard.

The range of the pitch bend is 2 semi-tones up or down by default but can be changed. To adjust the range of the pitch bend, open the MIDI configuration window by clicking on the **MIDI** button located just below the MIDI let in the top part of the interface and use the **Pitch Bend Range** drop-down list to select the range in semi-tones.

6.2.7 Modulation wheel

String Studio VS responds to MIDI modulation (MIDI controller number 1). For more details, please refer to Section 4.4.8.

7 Using *String Studio VS* as a Plug-In

String Studio VS is available in VST, RTAS, AAX and Audio Units formats and integrates seamlessly into the industry most popular multi-track recording and sequencing environments as a virtual instrument plug-in. *String Studio VS* works as any other plug-in in these environments so we recommend that you refer to your sequencer documentation in case you have problems running it as a plug-in. We review here some general points to keep in mind when using a plug-in version of *String Studio VS*.

7.1 Audio and MIDI Configuration

When *String Studio VS* is used as a plug-in, the audio and MIDI ports, sampling rate, buffer size, and audio format are determined by the host sequencer.

7.2 Automation

String Studio VS supports automation functions of host sequencers. All parameters visible on the interface can be automatized except for the **Polyphony**, **Bank**, **Program** and **History** commands.

7.3 Multiple Instances

Multiple instances of *String Studio VS* can be launched simultaneously in a host sequencer.

7.4 Saving Projects

When saving a project in a host sequencer, the currently loaded program is saved with the project in order to make sure that the instrument will be in the same state as when you saved the project when you re-open it. Note that banks of programs are not saved with the project which implies that if you are using MIDI program changes in your project, you must make sure that the bank you are using in your project still exists on your disk when you reload the project. The programs must also exist and be in the same order as when the project was saved.

7.5 Performance

Using a plug-in in a host sequencer requires CPU processing for both applications. The load on the CPU is even higher when multiple instances of a plug-in or numerous different plug-ins are used. To decrease CPU usage, remember that you can use the **freeze** or **bounce to track** functions of the host sequencer in order to render to audio the part played by a plug-in instead of recalculating it every time it is played.

8 License Agreement

IMPORTANT! CAREFULLY READ ALL THE TERMS AND CONDITIONS OF THIS AGREEMENT BEFORE OPENING THIS PACKAGE. OPENING THIS PACKAGE INDICATES YOUR ACCEPTANCE OF THESE TERMS AND CONDITIONS. IF YOU DO NOT AGREE WITH THE TERMS AND CONDITIONS OF THIS AGREEMENT, PROMPTLY RETURN THE UNOPENED PACKAGE AND ALL COMPONENTS THERETO TO THE PARTY FROM WHOM IT WAS ACQUIRED, FOR A FULL REFUND OF ANY CONSIDERATION PAID.

This software program, any printed materials, any on-line or electronic documentation, and any and all copies of such software program and materials (the "**Software**") are the copyrighted work of Applied Acoustics Systems DVM Inc. ("**AAS**"), its subsidiaries, licensors and/or its suppliers.

1. LICENSE TO USE. The Licensee is granted a personal, non-exclusive and non-transferable license to install and to use one copy of the Software on a single computer solely for the personal use of the Licensee. Use of the Software is subject to this Agreement.

2. RESTRICTIONS ON USE. The Licensee may not nor permit third parties to (i) make copies of any portion of the Software, other than as expressly permitted under this Agreement; (ii) modify, translate, disassemble, decompile, reverse engineer or create derivative and/or competitive products based on any portion of the Software; (iii) provide use of the Software in a network, timesharing, interactive cable television, multiple CPU service bureau or multiple user arrangement to users not individually licensed by AAS, other than as expressly permitted by the terms of this license. The Software is licensed to you as a single product. Its component parts may not be separated for use on more than one computer.

3. OWNERSHIP. AAS retains title to the Software, including but not limited to any titles, computer code, themes, objects dialog concepts, artwork, animations, sounds, audio effects, methods of operation, moral rights, any related documentation and "applets" incorporated into the Software. AAS retains ownership of and title to all intellectual property rights in the Software, underlying technology, related written materials, logos, names and other support materials furnished either with the Software or as a result of this Agreement, including but not limited to trade secrets, patents, trademarks and copyrights therein. Licensee shall not remove or alter any copyright or other proprietary rights notices contained on or within the Software and shall reproduce such notices on all copies thereof permitted under this Agreement or associated documentation.

4. LIMITED WARRANTY. Except for the foregoing, THE SOFTWARE IS provided "AS IS" without warranty or condition of any kind. AAS disclaims all warranties or conditions, written or oral, statutory, express or implied, including but not limited to the implied warranties of merchantable quality or fitness for a particular purpose, title and non-infringement of rights of any other person. AAS does not warrant that THE SOFTWARE will meet the Licensee's requirements or that the operation of the software will be uninterrupted or ERROR-FREE.

5. LIMITATION OF LIABILITY. TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, IN NO EVENT WILL AAS BE LIABLE TO THE LICENSEE OR ANY THIRD PARTY FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR EXEMPLARY DAMAGES WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOSS OF REVENUE OR PROFIT, LOST OR DAMAGED DATA, BUSINESS INTERRUPTION OR ANY OTHER PECUNIARY LOSS WHETHER BASED IN CONTRACT, TORT OR OTHER CAUSE OF ACTION, EVEN IF AAS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, EXCEPT IN RELATION TO GROSS NEGLIGENCE OR WILFUL BREACH OF THIS AGREEMENT BY AAS. NO AAS AGENT, REPRESENTATIVE OR DEALER IS AUTHORIZED TO EXTEND, MODIFY OR ADD TO THIS WARRANTY ON BEHALF OF AAS. THE TOTAL LIABILITY OF AAS FOR DAMAGES, WHETHER IN CONTRACT OR TORT, UNDER OR RELATED IN ANY WAY TO THIS AGREEMENT SHALL BE LIMITED TO THE LICENSE FEES ACTUALLY PAID BY LICENSEE TO AAS, OR IF NO FEES WERE PAID, AAS' LIST PRICE FOR THE SOFTWARE COVERED BY THIS LICENSE. THE EXCLUSION OF IMPLIED WARRANTIES AND/OR THE LIMITATION OF LIABILITY IS NOT PERMITTED IN SOME JURISDICTIONS, AND SOME OR ALL OF THESE EXCLUSIONS MAY THEREFORE NOT APPLY.

6. TERMINATION. This License also shall extend to the Software and any updates or new releases thereof obtained by the Licensee, if any, subject to any changes to this License made by AAS from time to time and provided to the Licensee, provided AAS is under a separate obligation to provide to Licensee such updates or upgrades and Licensee continues to have a valid license which is in effect at the time of receipt of each such update or new release. This License shall remain in effect until terminated. The Licensee may terminate this Agreement at any time, upon notification to AAS. This Agreement will terminate immediately without notice from AAS if the Licensee fails to comply with any provision of this License. Any such termination by AAS shall be in addition to and without prejudice to such rights and remedies as may be available, including injunction and other equitable remedies. Upon receipt of notice of termination from AAS, the Licensee must (a) immediately cease to use the Software; (b) destroy all copies of the Software, as well as copies of all documentation, specifications and magnetic media relating thereto in Licensee's possession or control; and (c) return all original versions of the Software and associated documentation. The provisions of Sections 1, 3, and 5 shall survive the termination of this Agreement.

7. GOVERNING LAW. This Agreement shall be governed by and construed in accordance with the laws of the Province of Quebec, without regard to the United Nations Convention On Contracts for the International Sale of Goods and conflict of laws provisions, if applicable, and the parties hereby irrevocably attorn to the jurisdiction of the courts of that province. Les parties sont d'accord à ce que cette convention soit rédigée en langue anglaise. The parties have agreed that this agreement be drafted in the English language.

8. SEVERABILITY. If any of the above provisions are held to be illegal, invalid or unenforceable, such provision shall be severed from this Agreement and this Agreement shall not be

rendered inoperative but the remaining provisions shall continue in full force and effect.

9. ENTIRE AGREEMENT. This Agreement is the entire agreement between AAS and the Licensee relating to the Software and: (i) supersedes all prior or contemporaneous oral or written communications, proposals and representations with respect to its subject matter; and (ii) prevails over any conflicting or additional terms of any quote, order, acknowledgement, or similar communication between the parties during the term of this Agreement except as otherwise expressly agreed by the parties. No modification to the Agreement will be binding, unless in writing and signed by a duly authorized representative of each party.

10. NON-WAIVER. No delay or failure to take any action or exercise any rights under this Agreement shall constitute a waiver or consent unless expressly waived or consented to in writing by a duly authorized representative of AAS. A waiver of any event does not apply to any other event, even if in relation to the same subject-matter.

Index

about, 57
adsr, 34
architecture, 13
arpeggiator, 52
    latch, 53
    looping, 53
    pattern, 52
    rate, 53
    rhythmic pattern, 53
    synchronization, 53
audio, 58
    configuration, 11, 58
    device, 58
    latency, 58
authorization, 8
auto-wah, 16, 36, 47

backup, 19
band-pass filter, 31, 32
bank, 17
    backup, 19
    create, 17
    delete, 17
    factory library, 20
    importing, 20
    rename, 17
    sharing, 19
bank manager, 17
body, 27
bow, 26
buffer size, 11, 58

challenge code, 8
chorus, 16, 36, 41
    mono, 41
    stereo, 41
clock, 23, 50
coarse, 22
compare, 56
compressor, 16, 36, 38
contact, 12
cpu load, 56

damper, 28
    sustain pedal, 28
default MIDI map, 59
delay, 16, 36, 40
distortion, 30, 40, 56
drop-down menus, 22

edit view, 15, 23
effect
    synchronization, 23
effect view, 16
effects
    master clock, 50
envelope, 34
EQ, 16, 37
equalizer, 16, 36
exciter, 24
    bow, 26
    hammer, 25
    plectrum, 24

factory library, 11
fade-in, 36
filter, 31, 34
    band-pass, 32
    band-pass filter, 31
    formant, 33
    formant filter, 31
    high-pass, 32
    high-pass filter, 31
    low-pass, 31
    low-pass filter, 31
    notch, 32
    notch filter, 31
fine tuning, 22
finger, 28
flanger, 16, 36, 42

formant filter, 31, 33
fret, 28
fx view, 16, 36

geometry, 29
glide, 51

hammer, 25
help, 10, 12
high-pass filter, 31, 32
history, 56

installation, 8
interface, 14, 22
    drop-down menus, 22
    knobs, 22
    numerical displays, 22
    switches, 22

key module, 50
keyboard, 50
    monophonic, 51
    polyphonic, 51
    priority, 51
    sustain pedal, 51
    unison, 51
knobs, 22

latency, 58
legato, 51
level meter, 56
lfo, 35
    fade-in, 36
    rate, 36
    synchronization, 23
    wave shape, 35
library
    importing, 20
    restore, 20
limiter, 56
low-pass filter, 31

manual, 57
master clock, 50
master volume, 56
micro-tuning, 55
MIDI, 58
    bank change, 60
    configuration, 11, 59
    device, 59
    link map, 59
    links, 59
    pitch bend, 53, 60
    program change, 17, 19, 60
    receiving, 55
MIDI controller, 59
MIDI led, 55
minimum system requirements, 7
modulation, 22, 35
modulation wheel, 14, 54
modules, 22
multi-effect, 16, 36
    auto wah, 47
    chorus, 41
    delay, 40
    distortion, 40
    EQ, 37
    flanger, 42
    notch filter, 47
    phaser, 45
    reverb, 48
    synchronization, 23
    wah wah, 47
multi-mode filter, 31

notch filter, 16, 31, 32, 36, 47
numerical displays, 22

overdrive, 40

parameters, 22
peak, 56
phaser, 16, 36, 45
pickup, 30
pitch bend, 14, 60
pitch wheel, 53
play view, 14, 50

plectrum, 24
plug-in, 12
    audio configuration, 61
    automation, 61
    formats, 61
    MIDI configuration, 61
    multiple instances, 61
    performance, 61
    saving projects, 61
polyphony, 8, 56
portamento, 51
programs, 11, 17
    backup, 19
    bank manager, 17
    copy, 17
    factory library, 20
    importing, 20
    move, 17
    saving, 17
    sharing, 19

registration, 8
response code, 8
restore factory library, 20
reverb, 16, 48
ribbon, 14, 54

sampling rate, 8, 11
save, 17
save as, 17
sc, 38
Scala files, 55
signal flow, 13
solid state, 40
soundboard, 27
standalone, 10
start-up, 10
string, 23
support, 12
sustain pedal, 28, 51
switches, 22
synchronization, 23, 50
synchronization to a host, 50
system requirements, 7

termination, 28
transpose, 55
tremolo, 36
tuning, 55

unlocking, 8
utility section, 55

vibrato, 14, 51
vintage chorus, 41
voices, 56
volume, 56
VU meter, 56

wah wah, 16, 36, 47
warm tube, 40